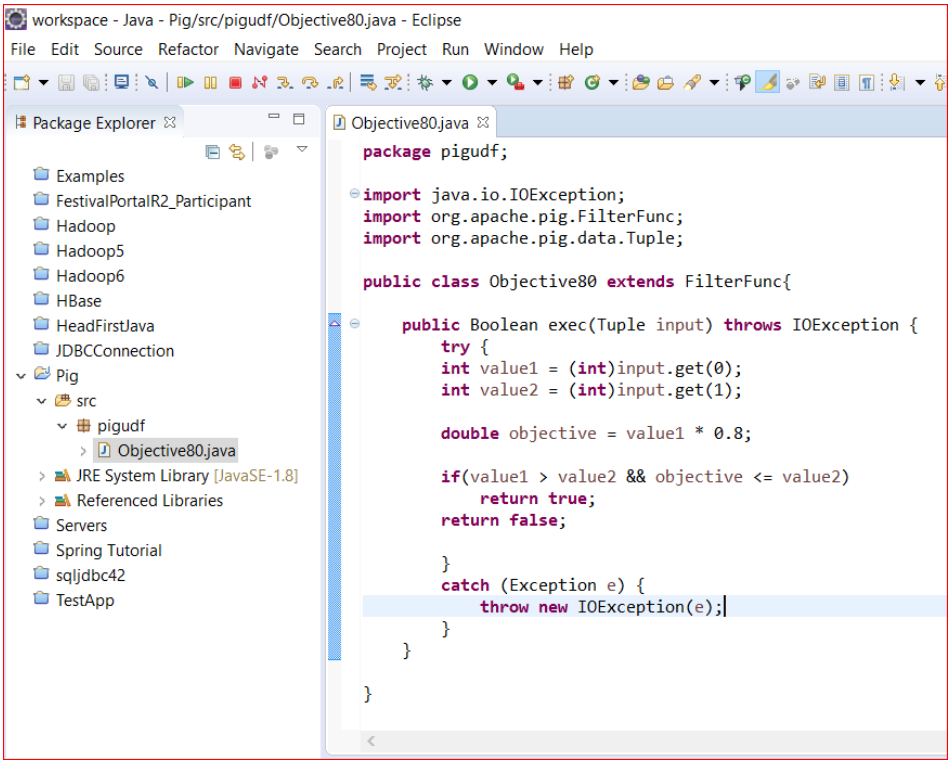
Problem Statement 2

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

Step 1: Write a java program for Pig UDF to filter data:



Note: Below jar files are required in order to compile pig IDF java program.

pig-0.8.3.jar

hadoop-common-2.6.0.jar

commons-logging-1.1.1.jar

Step 2: create a PigUDF.jar file of the project and move it to /home/acadgild/project2.1/

$ ll

```
[acadgild@localhost project2.1]$ ll
total 416
drwxrwxr-x. 3 acadgild acadgild   4096 Aug 25 16:48 flume_sink
-rw-rw-r--. 1 acadgild acadgild 391461 Aug 25 17:22 piggybank.jar
-rw-rw-r--. 1 acadgild acadgild   1654 Aug 25 19:48 PigUDF.jar
-rw-rw-r--. 1 acadgild acadgild   1434 Aug 25 18:09 problemStatement1.pig
-rw-rw-r--. 1 acadgild acadgild   1244 Aug 25 16:38 spool_dir.conf
-rw-rw-r--. 1 acadgild acadgild  14837 Aug 25 19:21 statewiseBPLacheived.java
[acadgild@localhost project2.1]$
```

Step 3: Start pig in mapreduce mode.

$ mr-jobhistory-daemon.sh start historyserver

$ pig

```
[acadgild@localhost project2.1]$ mr-jobhistory-daemon.sh start historyserver
starting historyserver, logging to /usr/local/hadoop-2.6.0/logs/mapred-acadgild-historyserver-localhost.localdomain.out
```

```
[acadgild@localhost project2.1]$ pig
2017-08-25 17:08:53,786 INFO  [main] pig.ExecTypeProvider: Trying ExecType : LOCAL
2017-08-25 17:08:53,789 INFO  [main] pig.ExecTypeProvider: Trying ExecType : MAPREDUCE
2017-08-25 17:08:53,789 INFO  [main] pig.ExecTypeProvider: Picked MAPREDUCE as the ExecType
2017-08-25 17:08:53,904 [main] INFO  org.apache.pig.Main - Apache Pig version 0.14.0 (r1640057) compiled Nov 16 2014, 18:02:05
2017-08-25 17:08:53,904 [main] INFO  org.apache.pig.Main - Logging error messages to: /home/acadgild/project2.1/pig_1503661133904.log
2017-08-25 17:08:53,951 [main] INFO  org.apache.pig.impl.util.Utils - Default bootup file /home/acadgild/.pigbootup not found
2017-08-25 17:08:54,378 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - mapred.job.tracker is deprecated. Instead, use mapreduce.jobtr
acker.address
2017-08-25 17:08:54,378 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - fs.default.name is deprecated. Instead, use fs.defaultFS
2017-08-25 17:08:54,378 [main] INFO  org.apache.pig.backend.hadoop.executionengine.HExecutionEngine - Connecting to hadoop file system at: hdfs://loca
lhost:9000
2017-08-25 17:08:54,383 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - mapred.used.genericoptionsparser is deprecated. Instead, use m
apreduce.client.genericoptionsparser.used
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/usr/local/hbase/lib/slf4j-log4j12-1.6.4.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/usr/local/hadoop-2.6.0/share/hadoop/common/lib/slf4j-log4j12-1.7.5.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
2017-08-25 17:08:54,689 [main] WARN  org.apache.hadoop.util.NativeCodeLoader - Unable to load native-hadoop library for your platform... using builtin
-java classes where applicable
2017-08-25 17:08:55,270 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - fs.default.name is deprecated. Instead, use fs.defaultFS
grunt>
```

Step 4: register piggybank.jar:

Copy file piggybank.jar to /home/acadgild/project2.1/

Piggybank.jar is need to be registered since we will be using below functions of the jar to load data into pig relation.

org.apache.pig.piggybank.storage.XMLLoader('XML_TAG'): this function is used for loading complete data under <XML_TAG> DATA </XML_TAG> in chararray datatype.

org.apache.pig.piggybank.evaluation.xml.XPath(chararray, 'XML_TAG/sub_XML_TAG'): This function is used to segregate the values mentioned between sub_XML_TAGS.

Below is the sample of the dataset:

```
<PhysicalProgress>
        <row>
                <State_Name>Andhra Pradesh</State_Name>
                <District_Name>ADILABAD</District_Name>
                <Project_Objectives_IHHL_BPL>247475</Project_Objectives_IHHL_BPL>
                <Project_Objectives_IHHL_APL>148181</Project_Objectives_IHHL_APL>
                <Project_Objectives_IHHL_TOTAL>395656</Project_Objectives_IHHL_TOTAL>
                <Project_Objectives_SCW>0</Project_Objectives_SCW>
                <Project_Objectives_School_Toilets>4462</Project_Objectives_School_Toilets>
                <Project_Objectives_Anganwadi_Toilets>427</Project_Objectives_Anganwadi_Toilets>
                <Project_Objectives_RSM>10</Project_Objectives_RSM>
                <Project_Objectives_PC>0</Project_Objectives_PC>
                <Project_Performance-IHHL_BPL>176300</Project_Performance-IHHL_BPL>
                <Project_Performance-IHHL_APL>52431</Project_Performance-IHHL_APL>
                <Project_Performance-IHHL_TOTAL>228731</Project_Performance-IHHL_TOTAL>
                <Project_Performance-SCW>0</Project_Performance-SCW>
                <Project_Performance-School_Toilets>4462</Project_Performance-School_Toilets>
                <Project_Performance-Anganwadi_Toilets>427</Project_Performance-Anganwadi_Toilets>
                <Project_Performance-RSM>0</Project_Performance-RSM>
                <Project_Performance-PC>0</Project_Performance-PC>
        </row>
        .
        .
        .
</PhysicalProgress>
```

grunt> REGISTER piggybank.jar;

grunt> DEFINE XPath org.apache.pig.piggybank.evaluation.xml.XPath();

Step 5: Register PigUDF.jar and define filter function Objective80:

grunt> REGISTER pigudf.jar

grunt> DEFINE ObjectiveFilter80 pigudf.Objective80;

```
grunt> REGISTER piggybank.jar
grunt> DEFINE XPath org.apache.pig.piggybank.evaluation.xml.XPath();
grunt> REGISTER pigudf.jar
grunt> DEFINE ObjectiveFilter80 pigudf.Objective80;
grunt>
```

Step 6: Load data into relation according to XML_TAG and sub_XML_TAGS:

grunt> A = LOAD '/flume_sink/' using org.apache.pig.piggybank.storage.XMLLoader('row') as (x:chararray);*

```
grunt> A = LOAD '/flume_sink/*' using org.apache.pig.piggybank.storage.XMLLoader('row') as (x:chararray);
2017-08-25 19:56:48,973 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - mapred.job.tracker.persist.jobstatus.hours is deprecated. Instead, use
 mapreduce.jobtracker.persist.jobstatus.hours
2017-08-25 19:56:48,973 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - mapred.heartbeats.in.second is deprecated. Instead, use mapreduce.jobt
racker.heartbeats.in.second
2017-08-25 19:56:48,973 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - jobclient.completion.poll.interval is deprecated. Instead, use mapredu
ce.client.completion.pollinterval
2017-08-25 19:56:48,973 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - mapred.tasktracker.tasks.sleeptime-before-sigkill is deprecated. Inste
```

grunt> B = FOREACH A GENERATE XPath(x, 'row/State_Name') AS state,

>> XPath(x, 'row/District_Name') AS dist,

>> XPath(x, 'row/Project_Objectives_IHHL_BPL') AS po_bpl,

>> XPath(x, 'row/Project_Objectives_IHHL_APL') AS po_apl,

>> XPath(x, 'row/Project_Objectives_IHHL_TOTAL') AS po_total,

>> XPath(x, 'row/Project_Objectives_SCW') AS po_scw,

>> XPath(x, 'row/Project_Objectives_School_Toilets') AS po_school_toilets,

>> XPath(x, 'row/Project_Objectives_Anganwadi_Toilets') AS po_anganwadi_toilets,

>> XPath(x, 'row/Project_Objectives_RSM') AS po_rsm,

>> XPath(x, 'row/Project_Objectives_PC') AS po_ps,

>> XPath(x, 'row/Project_Performance-IHHL_BPL') AS pp_bpl,

>> XPath(x, 'row/Project_Performance-IHHL_APL') AS pp_apl,

>> XPath(x, 'row/Project_Performance-IHHL_TOTAL') AS pp_total,

>> XPath(x, 'row/Project_Performance-SCW') AS pp_scw,

>> XPath(x, 'row/Project_Performance-School_Toilets') AS pp_school_toilets,

>> XPath(x, 'row/Project_Performance-Anganwadi_Toilets') AS pp_anganwadi_toilets,

>> XPath(x, 'row/Project_Performance-RSM') AS pp_rsm,

>> XPath(x, 'row/Project_Performance-PC') AS pp_pc;

```
grunt> B = FOREACH A GENERATE XPath(x, 'row/State_Name') AS state,
>> XPath(x, 'row/District_Name') AS dist,
>> XPath(x, 'row/Project_Objectives_IHHL_BPL') AS po_bpl,
>> XPath(x, 'row/Project_Objectives_IHHL_APL') AS po_apl,
>> XPath(x, 'row/Project_Objectives_IHHL_TOTAL') AS po_total,
>> XPath(x, 'row/Project_Objectives_SCW') AS po_scw,
>> XPath(x, 'row/Project_Objectives_School_Toilets') AS po_school_toilets,
>> XPath(x, 'row/Project_Objectives_Anganwadi_Toilets') AS po_anganwadi_toilets,
>> XPath(x, 'row/Project_Objectives_RSM') AS po_rsm,
>> XPath(x, 'row/Project_Objectives_PC') AS po_ps,
>> XPath(x, 'row/Project_Performance-IHHL_BPL') AS pp_bpl,
>> XPath(x, 'row/Project_Performance-IHHL_APL') AS pp_apl,
>> XPath(x, 'row/Project_Performance-IHHL_TOTAL') AS pp_total,
>> XPath(x, 'row/Project_Performance-SCW') AS pp_scw,
>> XPath(x, 'row/Project_Performance-School_Toilets') AS pp_school_toilets,
>> XPath(x, 'row/Project_Performance-Anganwadi_Toilets') AS pp_anganwadi_toilets,
>> XPath(x, 'row/Project_Performance-RSM') AS pp_rsm,
>> XPath(x, 'row/Project_Performance-PC') AS pp_pc;
grunt>
```

Step 7: take selected values which needs to be considered for further analysis and make them of required datatype.

grunt> C = FOREACH B GENERATE (chararray)state, (chararray)dist, (int)po_bpl, (int)pp_bpl;

grunt> describe C;

```
grunt> C = FOREACH B GENERATE (chararray)state, (chararray)dist, (int)po_bpl, (int)pp_bpl;
grunt> describe C;
C: {state: chararray,dist: chararray,po_bpl: int,pp_bpl: int}
grunt>
```

Step 9: filter data based on PigUDF function ObjectiveFilter80 (alias of pigudf.Objective80)

(rows having field ObjectiveFilter80's value as true will be saved in the relation E rest of the rows will be discarded)

grunt> D = FILTER C BY ObjectiveFilter80(po_bpl, pp_bpl);

grunt> describe D;

```
grunt> D = FILTER C BY ObjectiveFilter80(po_bpl, pp_bpl);
grunt> describe D;
D: {state: chararray,dist: chararray,po_bpl: int,pp_bpl: int}
grunt>
```

Step 10: Store relation D into HDFS directory:

grunt> STORE D INTO '/user/acadgild/project/StateWiseDevelopment/ProblemStatement2';

```
grunt> STORE D INTO '/user/acadgild/project/StateWiseDevelopment/ProblemStatement2';
2017-08-25 20:11:53,251 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - r
.counters.max
2017-08-25 20:11:53,251 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - 
um
2017-08-25 20:11:53,251 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - 
2017-08-25 20:11:53,286 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - r
```

```
2017-08-25 20:12:24,392 [main] INFO  org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MapReduceLauncher - 100% complete
2017-08-25 20:12:24,393 [main] INFO  org.apache.pig.tools.pigstats.mapreduce.SimplePigStats - Script Statistics:

HadoopVersion   PigVersion      UserId  StartedAt       FinishedAt      Features
2.2.0   0.14.0  acadgild        2017-08-25 20:11:53     2017-08-25 20:12:24     FILTER

Success!

Job Stats (time in seconds):
JobId   Maps    Reduces MaxMapTime      MinMapTime      AvgMapTime      MedianMapTime   MaxReduceTime   MinReduceTime   AvgReduceTime   MedianReducetime     A
lias    Feature Outputs
job_1503658924008_0010  1       0       15      15      15      15      0       0       0       0       A,B,C,D MAP_ONLY        /user/acadgild/project/StateWi
seDevelopment/ProblemStatement2,

Input(s):
Successfully read 0 records from: "/flume_sink/*"

Output(s):
Successfully stored 0 records in: "/user/acadgild/project/StateWiseDevelopment/ProblemStatement2"

Counters:
Total records written : 0
Total bytes written : 0
Spillable Memory Manager spill count : 0
Total bags proactively spilled: 0
Total records proactively spilled: 0

Job DAG:
job_1503658924008_0010


2017-08-25 20:12:24,397 [main] INFO  org.apache.hadoop.yarn.client.RMProxy - Connecting to ResourceManager at /0.0.0.0:8032
2017-08-25 20:12:24,399 [main] INFO  org.apache.hadoop.mapred.ClientServiceDelegate - Application state is completed. FinalApplicationStatus=SUCCEEDED. Redire
cting to job history server
2017-08-25 20:12:24,442 [main] INFO  org.apache.hadoop.yarn.client.RMProxy - Connecting to ResourceManager at /0.0.0.0:8032
2017-08-25 20:12:24,444 [main] INFO  org.apache.hadoop.mapred.ClientServiceDelegate - Application state is completed. FinalApplicationStatus=SUCCEEDED. Redire
cting to job history server
2017-08-25 20:12:24,476 [main] INFO  org.apache.hadoop.yarn.client.RMProxy - Connecting to ResourceManager at /0.0.0.0:8032
2017-08-25 20:12:24,490 [main] INFO  org.apache.hadoop.mapred.ClientServiceDelegate - Application state is completed. FinalApplicationStatus=SUCCEEDED. Redire
cting to job history server
2017-08-25 20:12:24,541 [main] WARN  org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MapReduceLauncher - Unable to retrieve job to compute warnin
g aggregation.
2017-08-25 20:12:24,541 [main] INFO  org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MapReduceLauncher - Success!
```

Step 11: check hdfs location where result have been stored:

$ hadoop fs -ls /user/acadgild/project/StateWiseDevelopment/ProblemStatement2

*$ hadoop fs -cat /user/acadgild/project/StateWiseDevelopment/ProblemStatement2/**

```
[acadgild@localhost project2.1]$ hadoop fs -ls /user/acadgild/project/StateWiseDevelopment/ProblemStatement2
17/08/25 20:15:01 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 2017-08-25 20:12 /user/acadgild/project/StateWiseDevelopment/ProblemStatement2/_SUCCESS
-rw-r--r--   1 acadgild supergroup       5670 2017-08-25 20:12 /user/acadgild/project/StateWiseDevelopment/ProblemStatement2/part-m-00000
[acadgild@localhost project2.1]$ hadoop fs -cat /user/acadgild/project/StateWiseDevelopment/ProblemStatement2/*
17/08/25 20:15:20 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Andhra Pradesh  CHITTOOR        296465  269750
Andhra Pradesh  CUDDAPAH        251653  239780
Andhra Pradesh  EAST GODAVARI   370255  347305
Andhra Pradesh  KRISHNA 351572  318730
Andhra Pradesh  KURNOOL 383478  323616
Andhra Pradesh  MEDAK   311743  310591
Andhra Pradesh  RANGAREDDI      212629  174460
Andhra Pradesh  WEST GODAVARI   344272  319477
Arunachal Pradesh       LOHIT   8800    8410
Assam   BAGSHA  85697   73500
Assam   CACHAR  119931  101075
Assam   DIBRUGARH       77606   69914
Assam   GOALPARA        65070   55747
Assam   GOLAGHAT        79743   77985
Assam   JORHAT  65070   52698
Assam   KAMRUP  85785   84725
Assam   KARIMGANJ       78992   75800
Assam   KOKRAJHAR       84830   75889
Assam   LAKHIMPUR       73224   60821
Assam   MARIGAON        60770   56490
Assam   NAGAON  168391  158290
```

PIG output have been stored successfully with the data separated by TAB (\t) which shows state,district,*Project_Objectives_IHHL_BPL,Project_Performance-IHHL_BPL* who achieved 80 percent **OR** more than 80% but less than 100% objective in BPL cards.

SQOOP: Export the results to mysql.

Step 1: start mysql/services:

$ sudo service mysqld status

$ sudo service mysqld start

$ sudo service mysqld status

```
[acadgild@localhost project2.1]$ sudo service mysqld status
[sudo] password for acadgild:
mysqld is stopped
[acadgild@localhost project2.1]$ sudo service mysqld start
Starting mysqld:                                           [  OK  ]
[acadgild@localhost project2.1]$ sudo service mysqld status
mysqld (pid  9463) is running...
[acadgild@localhost project2.1]$
```

$ mysql -u root

```
[acadgild@localhost project2.1]$ mysql -u root
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 2
Server version: 5.1.73 Source distribution

Copyright (c) 2000, 2013, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

Above command launches mysql with user root.

Step 2: create table statewiseBPLacheived along with the column details.

use db1;

show tables;

create table statewiseBPL80

(

state varchar(30),

dist varchar(30),

po_bpl int,

pp_bpl int

);

describe statewiseBPL80;

*select * from statewiseBPL80;*

```
mysql> use db1;
Database changed
mysql> show tables;
+----------------------+
| Tables_in_db1        |
+----------------------+
| customer             |
| statewiseBPLacheived |
+----------------------+
2 rows in set (0.00 sec)

mysql> create table statewiseBPL80
    -> (
    -> state varchar(30),
    -> dist varchar(30),
    -> po_bpl int,
    -> pp_bpl int
    -> );
Query OK, 0 rows affected (0.00 sec)

mysql> describe statewiseBPL80;
+--------+-------------+------+-----+---------+-------+
| Field  | Type        | Null | Key | Default | Extra |
+--------+-------------+------+-----+---------+-------+
| state  | varchar(30) | YES  |     | NULL    |       |
| dist   | varchar(30) | YES  |     | NULL    |       |
| po_bpl | int(11)     | YES  |     | NULL    |       |
| pp_bpl | int(11)     | YES  |     | NULL    |       |
+--------+-------------+------+-----+---------+-------+
4 rows in set (0.00 sec)

mysql> select * from statewiseBPL80;
Empty set (0.00 sec)

mysql>
```

Step 3: run sqoop export command to get data from output directory of the pig job to mysql table.

*sqoop export --connect jdbc:mysql://localhost/db1 *

*--username 'root' -P --table 'statewiseBPL80' *

*--export-dir '/user/acadgild/project/StateWiseDevelopment/ProblemStatement2/' *

*--input-fields-terminated-by '\t' *

-m 1

```
[acadgild@localhost project2.1]$ sqoop export --connect jdbc:mysql://localhost/db1 \
> --username 'root' -P --table 'statewiseBPL80' \
> --export-dir '/user/acadgild/project/StateWiseDevelopment/ProblemStatement2/' \
> --input-fields-terminated-by '\t' \
> -m 1
Warning: /usr/local/sqoop/../hcatalog does not exist! HCatalog jobs will fail.
Please set $HCAT_HOME to the root of your HCatalog installation.
Warning: /usr/local/sqoop/../accumulo does not exist! Accumulo imports will fail.
Please set $ACCUMULO_HOME to the root of your Accumulo installation.
Warning: /usr/local/sqoop/../zookeeper does not exist! Accumulo imports will fail.
Please set $ZOOKEEPER_HOME to the root of your Zookeeper installation.
2017-08-25 20:30:27,688 INFO  [main] sqoop.Sqoop: Running Sqoop version: 1.4.5
Enter password:
2017-08-25 20:30:29,732 INFO  [main] manager.MySQLManager: Preparing to use a MySQL streaming resultset.
2017-08-25 20:30:29,732 INFO  [main] tool.CodeGenTool: Beginning code generation
2017-08-25 20:30:30,041 INFO  [main] manager.SqlManager: Executing SQL statement: SELECT t.* FROM `statewiseBPL80` AS t LIMIT 1
2017-08-25 20:30:30,071 INFO  [main] manager.SqlManager: Executing SQL statement: SELECT t.* FROM `statewiseBPL80` AS t LIMIT 1
2017-08-25 20:30:30,080 INFO  [main] orm.CompilationManager: HADOOP_MAPRED_HOME is /usr/local/hadoop-2.6.0
Note: /tmp/sqoop-acadgild/compile/b38c5d3cacf1a46062ee4deab438dd93/statewiseBPL80.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.
```

```
2017-08-25 20:30:34,090 INFO  [main] mapreduce.JobSubmitter: Submitting tokens for job: job_1503658924008_0012
2017-08-25 20:30:34,448 INFO  [main] impl.YarnClientImpl: Submitted application application_1503658924008_0012 to Resource
2017-08-25 20:30:34,512 INFO  [main] mapreduce.Job: The url to track the job: http://http://localhost:8088/proxy/applicati
2017-08-25 20:30:34,513 INFO  [main] mapreduce.Job: Running job: job_1503658924008_0012
2017-08-25 20:30:41,649 INFO  [main] mapreduce.Job: Job job_1503658924008_0012 running in uber mode : false
2017-08-25 20:30:41,651 INFO  [main] mapreduce.Job:  map 0% reduce 0%
2017-08-25 20:30:46,717 INFO  [main] mapreduce.Job:  map 100% reduce 0%
2017-08-25 20:30:46,726 INFO  [main] mapreduce.Job: Job job_1503658924008_0012 completed successfully
```

Sqoop command completed successfully.

Step 4: check table in mysql:

*select * from statewiseBPL80;*

```
mysql> select * from statewiseBPL80;
+------------------+------------------------------+--------+--------+
| state            | dist                         | po_bpl | pp_bpl |
+------------------+------------------------------+--------+--------+
| Andhra Pradesh   | CHITTOOR                     | 296465 | 269750 |
| Andhra Pradesh   | CUDDAPAH                     | 251653 | 239780 |
| Andhra Pradesh   | EAST GODAVARI                | 370255 | 347305 |
| Andhra Pradesh   | KRISHNA                      | 351572 | 318730 |
| Andhra Pradesh   | KURNOOL                      | 383478 | 323616 |
| Andhra Pradesh   | MEDAK                        | 311743 | 310591 |
| Andhra Pradesh   | RANGAREDDI                   | 212629 | 174460 |
| Andhra Pradesh   | WEST GODAVARI                | 344272 | 319477 |
| Arunachal Pradesh | LOHIT                        |   8800 |   8410 |
| Assam            | BAGSHA                       |  85697 |  73500 |
| Assam            | CACHAR                       | 119931 | 101075 |
| Assam            | DIBRUGARH                    |  77606 |  69914 |
| Assam            | GOALPARA                     |  65070 |  55747 |
| Assam            | GOLAGHAT                     |  79743 |  77985 |
| Assam            | JORHAT                       |  65070 |  52698 |
| Assam            | KAMRUP                       |  85785 |  84725 |
| Assam            | KARIMGANJ                    |  78992 |  75800 |
| Assam            | KOKRAJHAR                    |  84830 |  75889 |
| Assam            | LAKHIMPUR                    |  73224 |  60821 |
| Assam            | MARIGAON                     |  60770 |  56490 |
| Assam            | NAGAON                       | 168391 | 158290 |
| Assam            | SIBSAGAR                     |  74686 |  63774 |
| Assam            | SONITPUR                     | 120048 | 101250 |
| Assam            | TINSUKIA                     |  68774 |  58808 |
| Bihar            | BEGUSARAI                    | 192877 | 168344 |
| Bihar            | MUZAFFARPUR                  | 333183 | 321510 |
| Bihar            | SAHARSA                      | 101440 |  83048 |
| Chhattisgarh     | DHAMTARI                     |  38501 |  32195 |
| Chhattisgarh     | JASHPUR                      |  56623 |  56014 |
| Chhattisgarh     | KANKER                       |  50835 |  41245 |
```

Step 5: Verify if all data have been exported from HDFS to MySQL:

Check number of lines in the HDFS file directory:

$ hadoop fs -cat /user/acadgild/project/StateWiseDevelopment/ProblemStatement2/ | wc -l*

```
[acadgild@localhost project2.1]$ hadoop fs -cat /user/acadgild/project/StateWiseDevelopment/ProblemStatement2/* | wc -l
17/08/25 20:33:32 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
172
[acadgild@localhost project2.1]$
```

Check the count of the Table statewiseBPLacheived in mysql:

select count(*) from statewiseBPL80;

```
mysql> select count(*) from statewiseBPL80;
+----------+
| count(*) |
+----------+
|      172 |
+----------+
1 row in set (0.00 sec)

mysql>
```

As compared above all the data has been exported from HDFS to mysql using Sqoop.

Store the results to HBase.

Step 1: Start HBase shell:

$ start-hbase.sh

$ hbase shell

```
[acadgild@localhost project2.1]$ start-hbase.sh
starting master, logging to /usr/local/hbase/logs/hbase-acadgild-master-localhost.localdomain.out
[acadgild@localhost project2.1]$ hbase shell
2017-08-25 20:43:08,389 INFO  [main] Configuration.deprecation: hadoop.native.lib is deprecated. Instead, use io.native.lib.available
HBase Shell; enter 'help<RETURN>' for list of supported commands.
Type "exit<RETURN>" to leave the HBase Shell
Version 0.98.14-hadoop2, r4e4aabb93b52f1b0fef6b66edd06ec8923014dec, Tue Aug 25 22:35:44 PDT 2015

hbase(main):001:0>
```

Step 2: create table statewiseBPLacheived with details as column family in hbase.

list

create 'statewiseBPLacheived','CF'

describe 'statewiseBPLacheived'

scan 'statewiseBPLacheived'

```
hbase(main):016:0> list
TABLE
clicks
customer
statewiseBPLacheived
3 row(s) in 0.0170 seconds

=> ["clicks", "customer", "statewiseBPLacheived"]
hbase(main):017:0> create 'statewiseBPL80','CF'
0 row(s) in 0.1530 seconds

=> Hbase::Table - statewiseBPL80
hbase(main):018:0> describe 'statewiseBPL80'
Table statewiseBPL80 is ENABLED
statewiseBPL80
COLUMN FAMILIES DESCRIPTION
{NAME => 'CF', BLOOMFILTER => 'ROW', VERSIONS => '1', IN_MEMORY => 'false', KEEP_DELETED_CELLS => 'FALSE', DATA_BLOCK_ENCODING => 'NONE', TTL => 'FOREVER', COMPRESSI
ON => 'NONE', MIN_VERSIONS => '0', BLOCKCACHE => 'true', BLOCKSIZE => '65536', REPLICATION_SCOPE => '0'}
1 row(s) in 0.0200 seconds

hbase(main):019:0> scan 'statewiseBPL80'
ROW                                      COLUMN+CELL
0 row(s) in 0.0080 seconds

hbase(main):020:0> count 'statewiseBPL80'
0 row(s) in 0.0070 seconds

=> 0
hbase(main):021:0>
```

Step 3: run below statements in pig mapreduce mode *to load data from HDFS file to pig relation:*

raw_data = LOAD '/user/acadgild/project/StateWiseDevelopment/ProblemStatement2/' USING PigStorage('\t') AS (*

state:chararray,

dist:chararray,

po_bpl:int,

pp_bpl:int

);

describe raw_data;

```
grunt> raw_data = LOAD '/user/acadgild/project/StateWiseDevelopment/ProblemStatement2/*' USING PigStorage('\t') AS (
>> state:chararray,
>> dist:chararray,
>> po_bpl:int,
>> pp_bpl:int
>> );
2017-08-25 21:59:28,425 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - mapreduce.job.counters.limit i
rs.max
2017-08-25 21:59:28,425 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - io.bytes.per.checksum is depre
2017-08-25 21:59:28,425 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - fs.default.name is deprecated.
grunt> describe raw_data;
raw_data: {state: chararray,dist: chararray,po_bpl: int,pp_bpl: int}
grunt>
```

Step 4: HBase stores data in the combination of ROWKEY and associated VALUES. Since we do not have any ROWKEY in above relation which consists of unique values for each records. Hence we will go ahead and create new column in the pig relation which will be the concatenation of STATE and DISTRICT. We have data of Project Objective and Project Performance associated with each STATE and DISTRICT hence if we concatenate these two column then resultant value will be unique to each record which can be used as ROWKEY for HBase table. Below pig command is used to create additional column with concatenation:

processed_data = FOREACH raw_data GENERATE CONCAT(state,dist) as rowkey, state, dist, po_bpl, pp_bpl;

describe processed_data;

```
grunt> processed_data = FOREACH raw_data GENERATE CONCAT(state,dist) as rowkey, state, dist, po_bpl, pp_bpl;
grunt> describe processed_data;
processed_data: {rowkey: chararray,state: chararray,dist: chararray,po_bpl: int,pp_bpl: int}
grunt>
```

Step 5: Store data in HBase table statewiseBPLacheived executing below pig command:

STORE processed_data INTO 'hbase://statewiseBPL80' USING org.apache.pig.backend.hadoop.hbase.HBaseStorage(

'CF:state,

CF:dist,

CF:po_bpl,

CF:pp_bpl'

);

```
2017-08-25 22:01:24,353 [main] INFO  org.apache.pig.tools.pigstats.mapreduce.SimplePigStats - Script Statistics:

HadoopVersion   PigVersion      UserId  StartedAt       FinishedAt      Features
2.2.0   0.14.0  acadgild        2017-08-25 22:01:06     2017-08-25 22:01:24     UNKNOWN

Success!

Job Stats (time in seconds):
JobId   Maps    Reduces MaxMapTime      MinMapTime      AvgMapTime      MedianMapTime   MaxReduceTime   MinReduceTime   AvgReduceTime   MedianReducetime        Alias
Feature Outputs
job_1503658924008_0022  1       0       4       4       4       4       0       0       0       0       processed_data,raw_data MAP_ONLY        hbase://statewiseBPL8
0,

Input(s):
Successfully read 0 records from: "/user/acadgild/project/StateWiseDevelopment/ProblemStatement2/*"

Output(s):
Successfully stored 0 records in: "hbase://statewiseBPL80"

Counters:
Total records written : 0
Total bytes written : 0
Spillable Memory Manager spill count : 0
Total bags proactively spilled: 0
Total records proactively spilled: 0

Job DAG:
job_1503658924008_0022


2017-08-25 22:01:24,360 [main] INFO  org.apache.hadoop.yarn.client.RMProxy - Connecting to ResourceManager at /0.0.0.0:8032
2017-08-25 22:01:24,369 [main] INFO  org.apache.hadoop.mapred.ClientServiceDelegate - Application state is completed. FinalApplicationStatus=SUCCEEDED. Redirecting t
o job history server
2017-08-25 22:01:24,451 [main] INFO  org.apache.hadoop.yarn.client.RMProxy - Connecting to ResourceManager at /0.0.0.0:8032
2017-08-25 22:01:24,463 [main] INFO  org.apache.hadoop.mapred.ClientServiceDelegate - Application state is completed. FinalApplicationStatus=SUCCEEDED. Redirecting t
o job history server
2017-08-25 22:01:24,507 [main] INFO  org.apache.hadoop.yarn.client.RMProxy - Connecting to ResourceManager at /0.0.0.0:8032
2017-08-25 22:01:24,514 [main] INFO  org.apache.hadoop.mapred.ClientServiceDelegate - Application state is completed. FinalApplicationStatus=SUCCEEDED. Redirecting t
o job history server
2017-08-25 22:01:24,566 [main] WARN  org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MapReduceLauncher - Unable to retrieve job to compute warning aggre
gation.
2017-08-25 22:01:24,566 [main] INFO  org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MapReduceLauncher - Success!
grunt>
```

File: H*Base_Load_statewiseBPL80.pig*

Step 6: scan HBase table:

scan 'statewiseBPL80'

```
hbase(main):021:0> scan 'statewiseBPL80'
ROW                                    COLUMN+CELL
 Andhra PradeshCHITTOOR                column=CF:dist, timestamp=1503678680330, value=CHITTOOR
 Andhra PradeshCHITTOOR                column=CF:po_bpl, timestamp=1503678680330, value=296465
 Andhra PradeshCHITTOOR                column=CF:pp_bpl, timestamp=1503678680330, value=269750
 Andhra PradeshCHITTOOR                column=CF:state, timestamp=1503678680330, value=Andhra Pradesh
 Andhra PradeshCUDDAPAH                column=CF:dist, timestamp=1503678680342, value=CUDDAPAH
 Andhra PradeshCUDDAPAH                column=CF:po_bpl, timestamp=1503678680342, value=251653
 Andhra PradeshCUDDAPAH                column=CF:pp_bpl, timestamp=1503678680342, value=239780
 Andhra PradeshCUDDAPAH                column=CF:state, timestamp=1503678680342, value=Andhra Pradesh
 Andhra PradeshEAST GODAVARI           column=CF:dist, timestamp=1503678680342, value=EAST GODAVARI
 Andhra PradeshEAST GODAVARI           column=CF:po_bpl, timestamp=1503678680342, value=370255
 Andhra PradeshEAST GODAVARI           column=CF:pp_bpl, timestamp=1503678680342, value=347305
 Andhra PradeshEAST GODAVARI           column=CF:state, timestamp=1503678680342, value=Andhra Pradesh
 Andhra PradeshKRISHNA                 column=CF:dist, timestamp=1503678680343, value=KRISHNA
 Andhra PradeshKRISHNA                 column=CF:po_bpl, timestamp=1503678680343, value=351572
 Andhra PradeshKRISHNA                 column=CF:pp_bpl, timestamp=1503678680343, value=318730
 Andhra PradeshKRISHNA                 column=CF:state, timestamp=1503678680343, value=Andhra Pradesh
 Andhra PradeshKURNOOL                 column=CF:dist, timestamp=1503678680343, value=KURNOOL
 Andhra PradeshKURNOOL                 column=CF:po_bpl, timestamp=1503678680343, value=383478
 Andhra PradeshKURNOOL                 column=CF:pp_bpl, timestamp=1503678680343, value=323616
 Andhra PradeshKURNOOL                 column=CF:state, timestamp=1503678680343, value=Andhra Pradesh
 Andhra PradeshMEDAK                   column=CF:dist, timestamp=1503678680343, value=MEDAK
 Andhra PradeshMEDAK                   column=CF:po_bpl, timestamp=1503678680343, value=311743
 Andhra PradeshMEDAK                   column=CF:pp_bpl, timestamp=1503678680343, value=310591
 Andhra PradeshMEDAK                   column=CF:state, timestamp=1503678680343, value=Andhra Pradesh
 Andhra PradeshRANGAREDDI              column=CF:dist, timestamp=1503678680344, value=RANGAREDDI
 Andhra PradeshRANGAREDDI              column=CF:po_bpl, timestamp=1503678680344, value=212629
 Andhra PradeshRANGAREDDI              column=CF:pp_bpl, timestamp=1503678680344, value=174460
 Andhra PradeshRANGAREDDI              column=CF:state, timestamp=1503678680344, value=Andhra Pradesh
 Andhra PradeshWEST GODAVARI           column=CF:dist, timestamp=1503678680344, value=WEST GODAVARI
 Andhra PradeshWEST GODAVARI           column=CF:po_bpl, timestamp=1503678680344, value=344272
 Andhra PradeshWEST GODAVARI           column=CF:pp_bpl, timestamp=1503678680344, value=319477
 Andhra PradeshWEST GODAVARI           column=CF:state, timestamp=1503678680344, value=Andhra Pradesh
```

Step 7: Verify if data is imported completely:

Check number of lines in the HDFS file directory:

$ hadoop fs -cat /user/acadgild/project/StateWiseDevelopment/ProblemStatement2/ | wc -l*

```
[acadgild@localhost project2.1]$ hadoop fs -cat /user/acadgild/project/StateWiseDevelopment/ProblemStatement2/* | wc -l
17/08/25 20:33:32 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
172
[acadgild@localhost project2.1]$
```

Count the number of rows in HBase table 'statewiseBPL80'

count 'statewiseBPL80'

```
hbase(main):022:0> count 'statewiseBPL80'
172 row(s) in 0.0370 seconds

=> 172
hbase(main):023:0>
```

As seen above all records have been imported to Hbase table successfully.

‘Store the results to HBase’ section is not a part of Project description/statement; this is solely for my understanding. Please do not reduce marks based on its evaluation however I would appreciate comments on the same. :)