Problem Statement 1*

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

Step 1: 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 2: 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();

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

```
grunt> REGISTER piggybank.jar;
grunt> DEFINE XPath org.apache.pig.piggybank.evaluation.xml.XPath();
grunt> A = LOAD '/flume_sink/*' using org.apache.pig.piggybank.storage.XMLLoader('row') as (x:chararray);
```

Step 2: Load data into relation according to sub_XML_TAGS:

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 3: 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 4: Create an hsdf directory /user/acadgild/project/StateWiseDevelopment/ProblemStatement1 to store the result:

$ hadoop fs -mkdir -p /user/acadgild/project/StateWiseDevelopment/

```
[acadgild@localhost project2.1]$ hadoop fs -mkdir -p /user/acadgild/project/StateWiseDevelopment/
17/08/25 17:49:22 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
[acadgild@localhost project2.1]$
```

Step 5: perform the pig logic to get the result and store it in above mentioned directory:

grunt> D = FILTER C BY po_bpl<=pp_bpl;

grunt> STORE D INTO '/user/acadgild/project/StateWiseDevelopment/ProblemStatement1';

```
grunt> D = FILTER C BY po_bpl<=pp_bpl;
grunt> STORE D INTO '/user/acadgild/project/StateWiseDevelopment/ProblemStatement1';
```

```
2017-08-25 17:57:14,465 [main] INFO  org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MapReduceLauncher - 100% complete
2017-08-25 17:57:14,465 [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 17:56:48     2017-08-25 17:57:14     FILTER

Success!

Job Stats (time in seconds):
JobId   Maps    Reduces MaxMapTime      MinMapTime      AvgMapTime      MedianMapTime   MaxReduceTime   MinReduceTime   AvgReduceTime   MedianReducetime        Alias  F
eature  Outputs
job_1503658924008_0005  1       0       14      14      14      14      0       0       0       0       A,B,C,D MAP_ONLY        /user/acadgild/project/StateWiseDevelopm
ent/ProblemStatement1,

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

Output(s):
Successfully stored 0 records in: "/user/acadgild/project/StateWiseDevelopment/ProblemStatement1"

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_0005


2017-08-25 17:57:14,470 [main] INFO  org.apache.hadoop.yarn.client.RMProxy - Connecting to ResourceManager at /0.0.0.0:8032
2017-08-25 17:57:14,479 [main] INFO  org.apache.hadoop.mapred.ClientServiceDelegate - Application state is completed. FinalApplicationStatus=SUCCEEDED. Redirecting to j
ob history server
2017-08-25 17:57:14,536 [main] INFO  org.apache.hadoop.yarn.client.RMProxy - Connecting to ResourceManager at /0.0.0.0:8032
2017-08-25 17:57:14,545 [main] INFO  org.apache.hadoop.mapred.ClientServiceDelegate - Application state is completed. FinalApplicationStatus=SUCCEEDED. Redirecting to j
ob history server
2017-08-25 17:57:14,587 [main] INFO  org.apache.hadoop.yarn.client.RMProxy - Connecting to ResourceManager at /0.0.0.0:8032
2017-08-25 17:57:14,589 [main] INFO  org.apache.hadoop.mapred.ClientServiceDelegate - Application state is completed. FinalApplicationStatus=SUCCEEDED. Redirecting to j
ob history server
2017-08-25 17:57:14,631 [main] WARN  org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MapReduceLauncher - Unable to retrieve job to compute warning aggregat
ion.
2017-08-25 17:57:14,631 [main] INFO  org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MapReduceLauncher - Success!
```

Mapreduce task run successfully.

Step 6: check hdfs location where result have been stored:

$ hadoop fs -ls /user/acadgild/project/StateWiseDevelopment/ProblemStatement1

*$ hadoop fs -cat /user/acadgild/project/StateWiseDevelopment/ProblemStatement1/**

```
[acadgild@localhost project2.1]$ hadoop fs -ls /user/acadgild/project/StateWiseDevelopment/ProblemStatement1
17/08/25 17:59:17 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 17:57 /user/acadgild/project/StateWiseDevelopment/ProblemStatement1/_SUCCESS
-rw-r--r--   1 acadgild supergroup       5980 2017-08-25 17:57 /user/acadgild/project/StateWiseDevelopment/ProblemStatement1/part-m-00000
[acadgild@localhost project2.1]$ hadoop fs -cat /user/acadgild/project/StateWiseDevelopment/ProblemStatement1/*
17/08/25 17:59:29 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Andhra Pradesh  ANANTAPUR       363314  366557
Andhra Pradesh  KARIMNAGAR      365267  369433
Andhra Pradesh  KHAMMAM 189225  195763
Andhra Pradesh  NALGONDA        215058  224813
Andhra Pradesh  NIZAMABAD       225519  225519
Andhra Pradesh  WARANGAL        330260  359732
Arunachal Pradesh       DIBANG VALLEY   1085    1088
Arunachal Pradesh       TIRAP   5780    5780
Assam   HAILAKANDI      49837   49837
Bihar   MADHUBANI       67482   67482
Bihar   VAISHALI        190598  196496
Chhattisgarh    KORBA   50691   63983
Goa     NORTH GOA       15000   15000
Gujarat AHMEDABAD       80192   80192
Gujarat BHAVNAGAR       31305   31563
Gujarat DANGS   27900   27900
Gujarat JAMNAGAR        45478   47822
Gujarat MAHESANA        61499   61938
Gujarat NAVSARI 75015   75015
Gujarat PATAN   58741   60002
Gujarat PORBANDAR       17024   17024
Gujarat RAJKOT  78753   79436
Gujarat SURAT   158797  158797
Gujarat VALSAD  85274   109073
Haryana BHIWANI 48947   49247
Haryana FARIDABAD       22254   22254
Haryana GURGAON 10522   14822
Haryana HISAR   46463   46463
Haryana JHAJJAR 22014   22014
Haryana KARNAL  45973   46015
```

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 100 percent 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 statewiseBPLacheived

(

state varchar(30),

dist varchar(30),

po_bpl int,

pp_bpl int

);

describe statewiseBPLacheived;

*select * from statewiseBPLacheived;*

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

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

mysql> describe statewiseBPLacheived;
+--------+-------------+------+-----+---------+-------+
| 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 statewiseBPLacheived;
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 'statewiseBPLacheived' \

--export-dir '/user/acadgild/project/StateWiseDevelopment/ProblemStatement1/' \

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

-m 1

```
[acadgild@localhost project2.1]$ sqoop export --connect jdbc:mysql://localhost/db1 \
> --username 'root' -P --table 'statewiseBPLacheived' \
> --export-dir '/user/acadgild/project/StateWiseDevelopment/ProblemStatement1/' \
> --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 19:21:29,753 INFO  [main] sqoop.Sqoop: Running Sqoop version: 1.4.5
Enter password:
2017-08-25 19:21:35,064 INFO  [main] manager.MySQLManager: Preparing to use a MySQL streaming resultset.
2017-08-25 19:21:35,065 INFO  [main] tool.CodeGenTool: Beginning code generation
2017-08-25 19:21:35,355 INFO  [main] manager.SqlManager: Executing SQL statement: SELECT t.* FROM `statewiseBPLacheived` AS t LIMIT 1
2017-08-25 19:21:35,380 INFO  [main] manager.SqlManager: Executing SQL statement: SELECT t.* FROM `statewiseBPLacheived` AS t LIMIT 1
2017-08-25 19:21:35,389 INFO  [main] orm.CompilationManager: HADOOP_MAPRED_HOME is /usr/local/hadoop-2.6.0
Note: /tmp/sqoop-acadgild/compile/d1f1686f936fdb49032c7dfe07cdf7e9/statewiseBPLacheived.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.
```

```
2017-08-25 19:21:42,578 INFO  [main] Configuration.deprecation: mapred.cache.files.filesizes is deprecated. Instead, use mapreduce.job.cache.files.filesizes
2017-08-25 19:21:42,742 INFO  [main] mapreduce.JobSubmitter: Submitting tokens for job: job_1503658924008_0007
2017-08-25 19:21:43,078 INFO  [main] impl.YarnClientImpl: Submitted application application_1503658924008_0007 to ResourceManager at /0.0.0.0:8032
2017-08-25 19:21:43,150 INFO  [main] mapreduce.Job: The url to track the job: http://http://localhost:8088/proxy/application_1503658924008_0007/
2017-08-25 19:21:43,151 INFO  [main] mapreduce.Job: Running job: job_1503658924008_0007
2017-08-25 19:21:50,260 INFO  [main] mapreduce.Job: Job job_1503658924008_0007 running in uber mode : false
2017-08-25 19:21:50,265 INFO  [main] mapreduce.Job:  map 0% reduce 0%
2017-08-25 19:21:55,338 INFO  [main] mapreduce.Job:  map 100% reduce 0%
2017-08-25 19:21:56,357 INFO  [main] mapreduce.Job: Job job_1503658924008_0007 completed successfully
```

Sqoop command completed successfully.

Step 4: check table in mysql:

*select * from statewiseBPLacheived;*

```
mysql> select * from statewiseBPLacheived;
+------------------+----------------------------+--------+--------+
| state            | dist                       | po_bpl | pp_bpl |
+------------------+----------------------------+--------+--------+
| Andhra Pradesh   | ANANTAPUR                  | 363314 | 366557 |
| Andhra Pradesh   | KARIMNAGAR                 | 365267 | 369433 |
| Andhra Pradesh   | KHAMMAM                    | 189225 | 195763 |
| Andhra Pradesh   | NALGONDA                   | 215058 | 224813 |
| Andhra Pradesh   | NIZAMABAD                  | 225519 | 225519 |
| Andhra Pradesh   | WARANGAL                   | 330260 | 359732 |
| Arunachal Pradesh | DIBANG VALLEY              |   1085 |   1088 |
| Arunachal Pradesh | TIRAP                      |   5780 |   5780 |
| Assam            | HAILAKANDI                 |  49837 |  49837 |
| Bihar            | MADHUBANI                  |  67482 |  67482 |
| Bihar            | VAISHALI                   | 190598 | 196496 |
| Chhattisgarh     | KORBA                      |  50691 |  63983 |
| Goa              | NORTH GOA                  |  15000 |  15000 |
| Gujarat          | AHMEDABAD                  |  80192 |  80192 |
| Gujarat          | BHAVNAGAR                  |  31305 |  31563 |
| Gujarat          | DANGS                      |  27900 |  27900 |
| Gujarat          | JAMNAGAR                   |  45478 |  47822 |
| Gujarat          | MAHESANA                   |  61499 |  61938 |
| Gujarat          | NAVSARI                    |  75015 |  75015 |
| Gujarat          | PATAN                      |  58741 |  60002 |
| Gujarat          | PORBANDAR                  |  17024 |  17024 |
| Gujarat          | RAJKOT                     |  78753 |  79436 |
| Gujarat          | SURAT                      | 158797 | 158797 |
| Gujarat          | VALSAD                     |  85274 | 109073 |
| Haryana          | BHIWANI                    |  48947 |  49247 |
| Haryana          | FARIDABAD                  |  22254 |  22254 |
| Haryana          | GURGAON                    |  10522 |  14822 |
| Haryana          | HISAR                      |  46463 |  46463 |
| Haryana          | JHAJJAR                    |  22014 |  22014 |
| Haryana          | KARNAL                     |  45973 |  46015 |
| Haryana          | KURUKSHETRA                |  30598 |  30681 |
| Haryana          | MAHENDRAGARH               |  17500 |  17500 |
| Haryana          | PANCHKULA                  |   8760 |   8760 |
| Haryana          | PANIPAT                    |  28000 |  28000 |
| Haryana          | ROHTAK                     |  22171 |  22171 |
| Haryana          | SIRSA                      |  35400 |  35400 |
| Haryana          | SONIPAT                    |  29808 |  30300 |
| Himachal Pradesh | BILASPUR                   |  11931 |  13078 |
| Himachal Pradesh | CHAMBA                     |  44429 |  58422 |
```

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/ProblemStatement1/ | wc -l*

```
[acadgild@localhost project2.1]$ hadoop fs -cat /user/acadgild/project/StateWiseDevelopment/ProblemStatement1/* | wc -l
17/08/25 19:30:32 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
176
[acadgild@localhost project2.1]$
```

Check the count of the Table statewiseBPLacheived in mysql:

select count(*) from statewiseBPLacheived;

```
mysql> select count(*) from statewiseBPLacheived;
+----------+
| count(*) |
+----------+
|      176 |
+----------+
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):001:0> list
TABLE
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/Stat
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
2017-08-25 20:50:40,966 WARN  [main] util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using bu
ble
clicks
customer
2 row(s) in 1.3640 seconds

=> ["clicks", "customer"]
hbase(main):002:0> create 'statewiseBPLacheived','CF'
0 row(s) in 0.2430 seconds

=> Hbase::Table - statewiseBPLacheived
hbase(main):003:0> describe 'statewiseBPLacheived'
Table statewiseBPLacheived is ENABLED
statewiseBPLacheived
COLUMN FAMILIES DESCRIPTION
{NAME => 'CF', BLOOMFILTER => 'ROW', VERSIONS => '1', IN_MEMORY => 'false', KEEP_DELETED_CELLS => 'FALSE', DATA_BLOCK_ENCODING
 => 'NONE', TTL => 'FOREVER', COMPRESSION => 'NONE', MIN_VERSIONS => '0', BLOCKCACHE => 'true', BLOCKSIZE => '65536', REPLICAT
ION_SCOPE => '0'}
1 row(s) in 0.0740 seconds
```

```
hbase(main):006:0> scan 'statewiseBPLacheived'
ROW                                   COLUMN+CELL
0 row(s) in 0.0640 seconds

hbase(main):007:0>
```

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/ProblemStatement1/' 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/ProblemStatement1/*' USING PigStorage('\t') AS (
>> state:chararray,
>> dist:chararray,
>> po_bpl:int,
>> pp_bpl:int
>> );
2017-08-25 21:37:27,628 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - mapreduce.job.counters.limit is deprecated. Instead, use mapreduce.job.counte
rs.max
2017-08-25 21:37:27,632 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - io.bytes.per.checksum is deprecated. Instead, use dfs.bytes-per-checksum
2017-08-25 21:37:27,632 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - fs.default.name is deprecated. Instead, use fs.defaultFS
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://statewiseBPLacheived' USING org.apache.pig.backend.hadoop.hbase.HBaseStorage(

'CF:state,

CF:dist,

CF:po_bpl,

CF:pp_bpl'

);

```
2017-08-25 21:48:24,447 [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 21:48:08     2017-08-25 21:48:24     UNKNOWN

Success!

Job Stats (time in seconds):
JobId   Maps    Reduces MaxMapTime      MinMapTime      AvgMapTime      MedianMapTime   MaxReduceTime   MinReduceTime   AvgReduceTime   MedianReducetime        Alias
Feature Outputs
job_1503658924008_0021  1       0       4       4       4       4       0       0       0       0       processed_data,raw_data MAP_ONLY        hbase://statewiseBPLa
cheived,

Input(s):
Successfully read 0 records from: "/user/acadgild/project/StateWiseDevelopment/ProblemStatement1/*"

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

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_0021


2017-08-25 21:48:24,452 [main] INFO  org.apache.hadoop.yarn.client.RMProxy - Connecting to ResourceManager at /0.0.0.0:8032
2017-08-25 21:48:24,456 [main] INFO  org.apache.hadoop.mapred.ClientServiceDelegate - Application state is completed. FinalApplicationStatus=SUCCEEDED. Redirecting t
o job history server
2017-08-25 21:48:24,530 [main] INFO  org.apache.hadoop.yarn.client.RMProxy - Connecting to ResourceManager at /0.0.0.0:8032
2017-08-25 21:48:24,537 [main] INFO  org.apache.hadoop.mapred.ClientServiceDelegate - Application state is completed. FinalApplicationStatus=SUCCEEDED. Redirecting t
o job history server
2017-08-25 21:48:24,578 [main] INFO  org.apache.hadoop.yarn.client.RMProxy - Connecting to ResourceManager at /0.0.0.0:8032
2017-08-25 21:48:24,580 [main] INFO  org.apache.hadoop.mapred.ClientServiceDelegate - Application state is completed. FinalApplicationStatus=SUCCEEDED. Redirecting t
o job history server
2017-08-25 21:48:24,619 [main] WARN  org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MapReduceLauncher - Unable to retrieve job to compute warning aggre
gation.
2017-08-25 21:48:24,619 [main] INFO  org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MapReduceLauncher - Success!
grunt>
```


File: HBase_Load_*statewiseBPLacheived.pig*

Step 6: scan HBase table:

scan 'statewiseBPLacheived'

```
hbase(main):014:0> scan 'statewiseBPLacheived'
ROW                                   COLUMN+CELL
 Andhra PradeshANANTAPUR              column=CF:dist, timestamp=1503677900220, value=ANANTAPUR
 Andhra PradeshANANTAPUR              column=CF:po_bpl, timestamp=1503677900220, value=363314
 Andhra PradeshANANTAPUR              column=CF:pp_bpl, timestamp=1503677900220, value=366557
 Andhra PradeshANANTAPUR              column=CF:state, timestamp=1503677900220, value=Andhra Pradesh
 Andhra PradeshKARIMNAGAR             column=CF:dist, timestamp=1503677900231, value=KARIMNAGAR
 Andhra PradeshKARIMNAGAR             column=CF:po_bpl, timestamp=1503677900231, value=365267
 Andhra PradeshKARIMNAGAR             column=CF:pp_bpl, timestamp=1503677900231, value=369433
 Andhra PradeshKARIMNAGAR             column=CF:state, timestamp=1503677900231, value=Andhra Pradesh
 Andhra PradeshKHAMMAM                column=CF:dist, timestamp=1503677900231, value=KHAMMAM
 Andhra PradeshKHAMMAM                column=CF:po_bpl, timestamp=1503677900231, value=189225
 Andhra PradeshKHAMMAM                column=CF:pp_bpl, timestamp=1503677900231, value=195763
 Andhra PradeshKHAMMAM                column=CF:state, timestamp=1503677900231, value=Andhra Pradesh
 Andhra PradeshNALGONDA               column=CF:dist, timestamp=1503677900232, value=NALGONDA
 Andhra PradeshNALGONDA               column=CF:po_bpl, timestamp=1503677900232, value=215058
 Andhra PradeshNALGONDA               column=CF:pp_bpl, timestamp=1503677900232, value=224813
 Andhra PradeshNALGONDA               column=CF:state, timestamp=1503677900232, value=Andhra Pradesh
 Andhra PradeshNIZAMABAD              column=CF:dist, timestamp=1503677900232, value=NIZAMABAD
 Andhra PradeshNIZAMABAD              column=CF:po_bpl, timestamp=1503677900232, value=225519
 Andhra PradeshNIZAMABAD              column=CF:pp_bpl, timestamp=1503677900232, value=225519
 Andhra PradeshNIZAMABAD              column=CF:state, timestamp=1503677900232, value=Andhra Pradesh
 Andhra PradeshWARANGAL               column=CF:dist, timestamp=1503677900232, value=WARANGAL
 Andhra PradeshWARANGAL               column=CF:po_bpl, timestamp=1503677900232, value=330260
 Andhra PradeshWARANGAL               column=CF:pp_bpl, timestamp=1503677900232, value=359732
 Andhra PradeshWARANGAL               column=CF:state, timestamp=1503677900232, value=Andhra Pradesh
 Arunachal PradeshDIBANG VALLEY       column=CF:dist, timestamp=1503677900232, value=DIBANG VALLEY
 Arunachal PradeshDIBANG VALLEY       column=CF:po_bpl, timestamp=1503677900232, value=1085
 Arunachal PradeshDIBANG VALLEY       column=CF:pp_bpl, timestamp=1503677900232, value=1088
 Arunachal PradeshDIBANG VALLEY       column=CF:state, timestamp=1503677900232, value=Arunachal Pradesh
 Arunachal PradeshTIRAP               column=CF:dist, timestamp=1503677900233, value=TIRAP
 Arunachal PradeshTIRAP               column=CF:po_bpl, timestamp=1503677900233, value=5780
 Arunachal PradeshTIRAP               column=CF:pp_bpl, timestamp=1503677900233, value=5780
 Arunachal PradeshTIRAP               column=CF:state, timestamp=1503677900233, value=Arunachal 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/ProblemStatement1/ | wc -l*

```
[acadgild@localhost project2.1]$ hadoop fs -cat /user/acadgild/project/StateWiseDevelopment/ProblemStatement1/* | wc -l
17/08/25 19:30:32 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
176
[acadgild@localhost project2.1]$
```

Count the number of rows in HBase table statewiseBPLacheived

```
hbase(main):015:0> count 'statewiseBPLacheived'
176 row(s) in 0.0510 seconds

=> 176
hbase(main):016:0>
```

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. :)