Big Data Hadoop and Spark Development

Session 7: EXPLORING APACHE PIG

Assignment

Task 1

Write a program to implement wordcount using Pig.

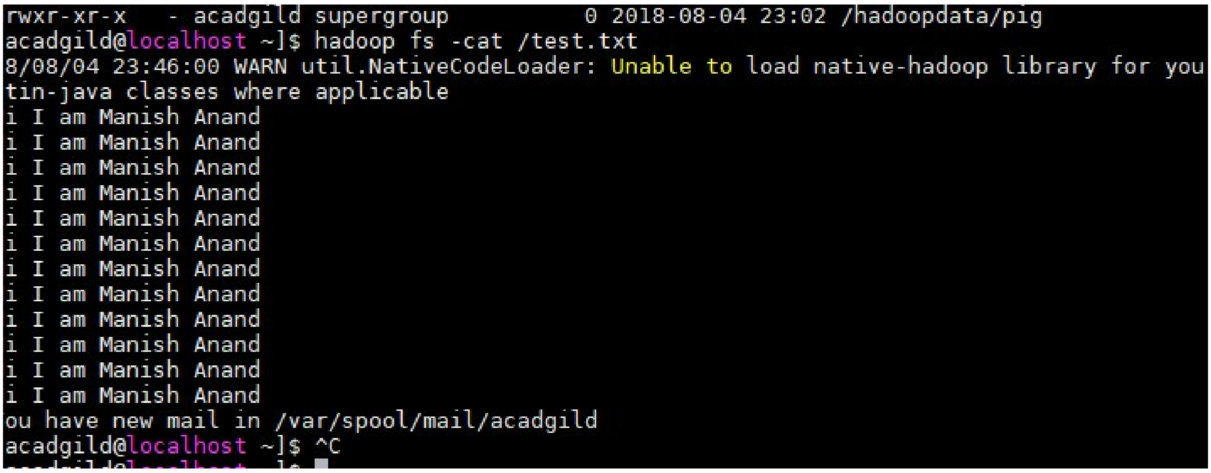
A = load '/test.txt';

B = foreach A generate flatten(TOKENIZE((chararray)$0)) as word;

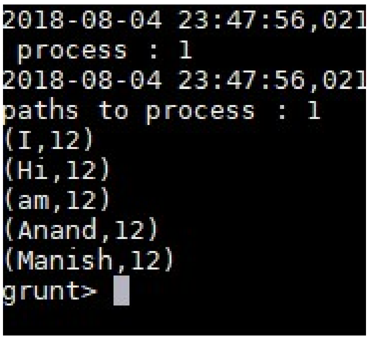
C = group B by word;

D = foreach C generate group, COUNT(B);

dump D;



```
rwxr-xr-x   - acadgild supergroup          0 2018-08-04 23:02 /hadoopdata/pig
acadgild@localhost ~]$ hadoop fs -cat /test.txt
8/08/04 23:46:00 WARN util.NativeCodeLoader: Unable to load native-hadoop library for you
tin-java classes where applicable
i I am Manish Anand
i I am Manish Anand
i I am Manish Anand
i I am Manish Anand
i I am Manish Anand
i I am Manish Anand
i I am Manish Anand
i I am Manish Anand
i I am Manish Anand
i I am Manish Anand
i I am Manish Anand
i I am Manish Anand
ou have new mail in /var/spool/mail/acadgild
acadgild@localhost ~]$ ^C
```



```
2018-08-04 23:47:56,021
 process : 1
2018-08-04 23:47:56,021
paths to process : 1
(I,12)
(Hi,12)
(am,12)
(Anand,12)
(Manish,12)
grunt>
```

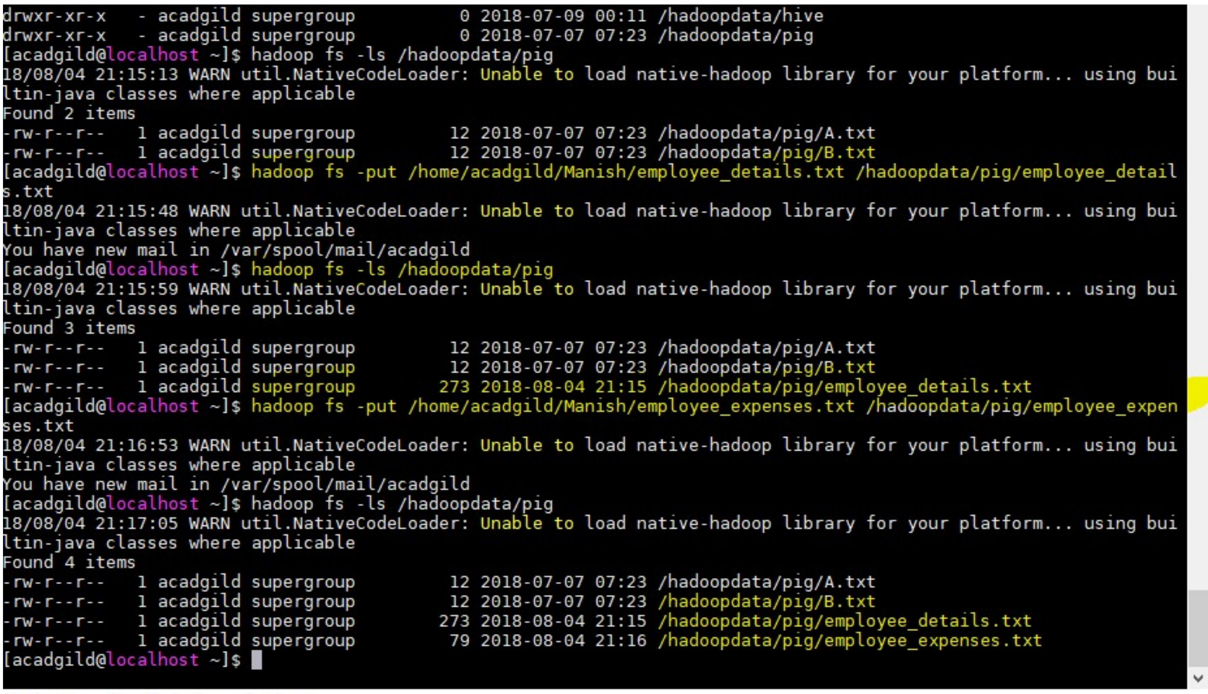
Task 2

We have employee_details and employee_expenses files. Use local mode while running Pig and write Pig Latin script to get below results:

employee_details (EmpID,Name,Salary,EmployeeRating)
https://github.com/prateekATacadgild/DatasetsForCognizant/blob/master/employee_details.txt

employee_expenses(EmpID,Expence)
https://github.com/prateekATacadgild/DatasetsForCognizant/blob/master/employee_expenses.txt

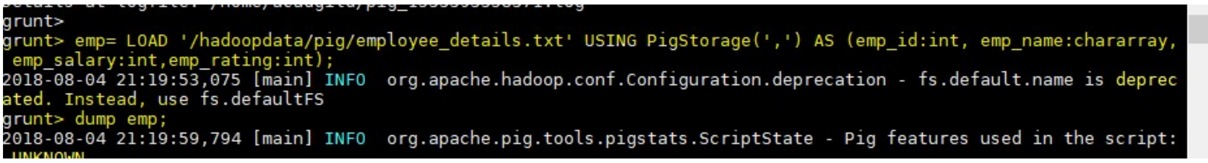
Step :1 Put the both the files in HDFS



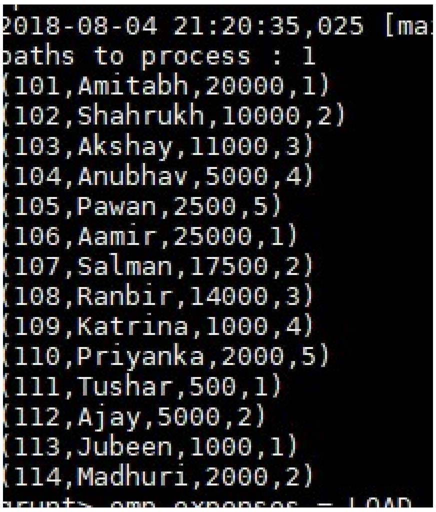
```
drwxr-xr-x   - acadgild supergroup          0 2018-07-09 00:11 /hadoopdata/hive
drwxr-xr-x   - acadgild supergroup          0 2018-07-07 07:23 /hadoopdata/pig
[acadgild@localhost ~]$ hadoop fs -ls /hadoopdata/pig
18/08/04 21:15:13 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using bui
ltin-java classes where applicable
Found 2 items
-rw-r--r--   1 acadgild supergroup         12 2018-07-07 07:23 /hadoopdata/pig/A.txt
-rw-r--r--   1 acadgild supergroup         12 2018-07-07 07:23 /hadoopdata/pig/B.txt
[acadgild@localhost ~]$ hadoop fs -put /home/acadgild/Manish/employee_details.txt /hadoopdata/pig/employee_detail
s.txt
18/08/04 21:15:48 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using bui
ltin-java classes where applicable
You have new mail in /var/spool/mail/acadgild
[acadgild@localhost ~]$ hadoop fs -ls /hadoopdata/pig
18/08/04 21:15:59 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using bui
ltin-java classes where applicable
Found 3 items
-rw-r--r--   1 acadgild supergroup         12 2018-07-07 07:23 /hadoopdata/pig/A.txt
-rw-r--r--   1 acadgild supergroup         12 2018-07-07 07:23 /hadoopdata/pig/B.txt
-rw-r--r--   1 acadgild supergroup        273 2018-08-04 21:15 /hadoopdata/pig/employee_details.txt
[acadgild@localhost ~]$ hadoop fs -put /home/acadgild/Manish/employee_expenses.txt /hadoopdata/pig/employee_expen
ses.txt
18/08/04 21:16:53 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using bui
ltin-java classes where applicable
You have new mail in /var/spool/mail/acadgild
[acadgild@localhost ~]$ hadoop fs -ls /hadoopdata/pig
18/08/04 21:17:05 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using bui
ltin-java classes where applicable
Found 4 items
-rw-r--r--   1 acadgild supergroup         12 2018-07-07 07:23 /hadoopdata/pig/A.txt
-rw-r--r--   1 acadgild supergroup         12 2018-07-07 07:23 /hadoopdata/pig/B.txt
-rw-r--r--   1 acadgild supergroup        273 2018-08-04 21:15 /hadoopdata/pig/employee_details.txt
-rw-r--r--   1 acadgild supergroup         79 2018-08-04 21:16 /hadoopdata/pig/employee_expenses.txt
[acadgild@localhost ~]$
```

(a) Top 5 employees (employee id and employee name) with highest rating. (In case two employees have same rating, employee with name coming first in dictionary should get preference)

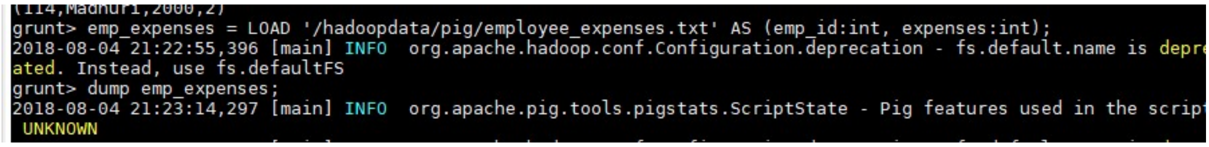
Step1: Load the file



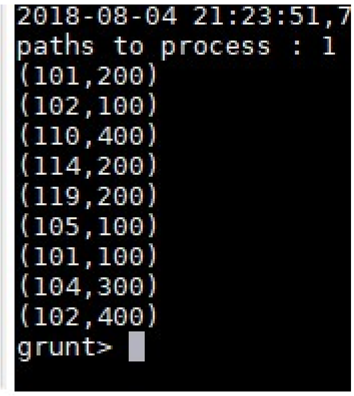
```
grunt>
grunt> emp= LOAD '/hadoopdata/pig/employee_details.txt' USING PigStorage(',') AS (emp_id:int, emp_name:chararray,
 emp_salary:int,emp_rating:int);
2018-08-04 21:19:53,075 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - fs.default.name is deprec
ated. Instead, use fs.defaultFS
grunt> dump emp;
2018-08-04 21:19:59,794 [main] INFO  org.apache.pig.tools.pigstats.ScriptState - Pig features used in the script:
 UNKNOWN
```



```
2018-08-04 21:20:35,025 [ma
paths to process : 1
(101,Amitabh,20000,1)
(102,Shahrukh,10000,2)
(103,Akshay,11000,3)
(104,Anubhav,5000,4)
(105,Pawan,2500,5)
(106,Aamir,25000,1)
(107,Salman,17500,2)
(108,Ranbir,14000,3)
(109,Katrina,1000,4)
(110,Priyanka,2000,5)
(111,Tushar,500,1)
(112,Ajay,5000,2)
(113,Jubeen,1000,1)
(114,Madhuri,2000,2)
grunt> emp_expenses = LOAD
```



```
(114,Madhuri,2000,2)
grunt> emp_expenses = LOAD '/hadoopdata/pig/employee_expenses.txt' AS (emp_id:int, expenses:int);
2018-08-04 21:22:55,396 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - fs.default.name is depre
ated. Instead, use fs.defaultFS
grunt> dump emp_expenses;
2018-08-04 21:23:14,297 [main] INFO  org.apache.pig.tools.pigstats.ScriptState - Pig features used in the script
 UNKNOWN
```



```
2018-08-04 21:23:51,7
paths to process : 1
(101,200)
(102,100)
(110,400)
(114,200)
(119,200)
(105,100)
(101,100)
(104,300)
(102,400)
grunt>
```

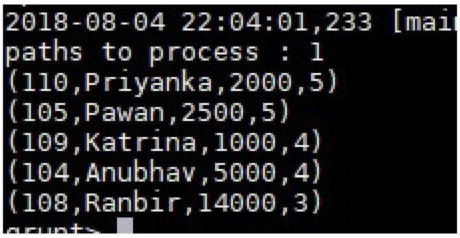
Pig Query:

emp= LOAD '/hadoopdata/pig/employee_details.txt' USING PigStorage(',') AS (emp_id:int, emp_name:chararray, emp_salary:int,emp_rating:int);

rating = order emp by emp_rating DESC;

Result = LIMIT rating 5;

Dump Result;

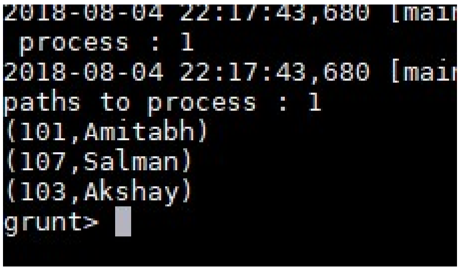


```
2018-08-04 22:04:01,233 [mai
paths to process : 1
(110,Priyanka,2000,5)
(105,Pawan,2500,5)
(109,Katrina,1000,4)
(104,Anubhav,5000,4)
(108,Ranbir,14000,3)
grunt>
```

(b) Top 3 employees (employee id and employee name) with highest salary, whose employee id is an odd number. (In case two employees have same salary, employee with name coming first in dictionary should get preference)

Pig Query:

```
emp= LOAD 'employee_details.txt' USING PigStorage(',') AS (emp_id:int,
emp_name:chararray, emp_salary:int,emp_rating:int);

 emp_sal_name = order emp by emp_salary desc;

 emp_sal_id = FILTER emp_sal_name by emp_id%2==1;

 emp_final = FOREACH emp_sal_id generate  emp_id,emp_name;

 emp_final_limit = LIMIT emp_final 3;
```



```
2018-08-04 22:17:43,680 [mai
 process : 1
2018-08-04 22:17:43,680 [mai
paths to process : 1
(101,Amitabh)
(107,Salman)
(103,Akshay)
grunt> 
```

(c) Employee (employee id and employee name) with maximum expense (In case two employees have same expense, employee with name coming first in dictionary should get preference)

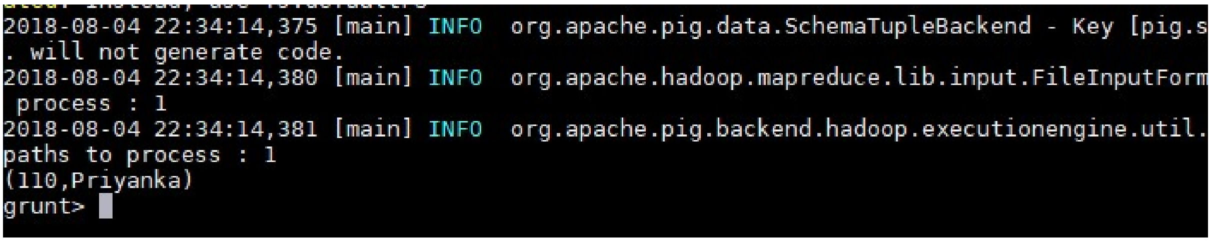
Pig_Query:

```
emp = LOAD 'employee_details.txt' USING PigStorage(',') AS (emp_id:int,
emp_name:chararray, emp_salary:int);

 emp_expenses = LOAD '/hadoopdata/pig/employee_expenses.txt' AS (emp_id:int,
expenses:int);

 Joinempempexpense = join emp by emp_id, emp_expenses by emp_id;

maxexpense = ORDER Joinempempexpense by  emp_expenses::expenses desc;
```

Limitmaxepnse = LIMIT maxexpense 1;

Limitmaxexpensefinal = foreach Limitmaxepnse generate emp::emp_id,emp::emp_name;

dump Limitmaxexpensefinal;



```
2018-08-04 22:34:14,375 [main] INFO  org.apache.pig.data.SchemaTupleBackend - Key [pig.s
. will not generate code.
2018-08-04 22:34:14,380 [main] INFO  org.apache.hadoop.mapreduce.lib.input.FileInputForm
 process : 1
2018-08-04 22:34:14,381 [main] INFO  org.apache.pig.backend.hadoop.executionengine.util.
paths to process : 1
(110,Priyanka)
grunt>
```

(d) List of employees (employee id and employee name) having entries in employee_expenses file.

Pig Query:

emp= LOAD 'employee_details.txt' USING PigStorage(',') AS (emp_id:int, emp_name:chararray, emp_salary:int,emp_rating:int);

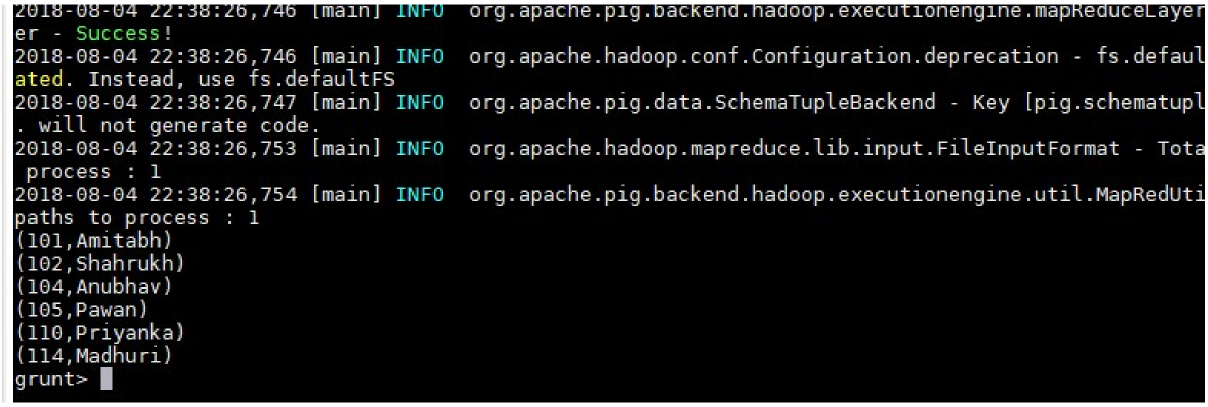
emp_expenses = LOAD '/hadoopdata/pig/employee_expenses.txt' AS (emp_id:int, expenses:int);

emp_with_exp = JOIN emp BY emp_id, emp_expenses BY emp_id;

emp_with_exp_data = FOREACH emp_with_exp GENERATE emp::emp_id, emp::emp_name;

emp_with_exp_distinct_data = DISTINCT emp_with_exp_data;

dump emp_with_exp_distinct_data;



```
2018-08-04 22:38:26,746 [main] INFO  org.apache.pig.backend.hadoop.executionengine.mapReduceLayer
er - Success!
2018-08-04 22:38:26,746 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - fs.defaul
ated. Instead, use fs.defaultFS
2018-08-04 22:38:26,747 [main] INFO  org.apache.pig.data.SchemaTupleBackend - Key [pig.schematupl
. will not generate code.
2018-08-04 22:38:26,753 [main] INFO  org.apache.hadoop.mapreduce.lib.input.FileInputFormat - Tota
 process : 1
2018-08-04 22:38:26,754 [main] INFO  org.apache.pig.backend.hadoop.executionengine.util.MapRedUti
paths to process : 1
(101,Amitabh)
(102,Shahrukh)
(104,Anubhav)
(105,Pawan)
(110,Priyanka)
(114,Madhuri)
grunt>
```

(e) List of employees (employee id and employee name) having no entry in employee_expenses file.

Pig Query:

emp= LOAD 'employee_details.txt' USING PigStorage(',') AS (emp_id:int, emp_name:chararray, emp_salary:int,emp_rating:int);

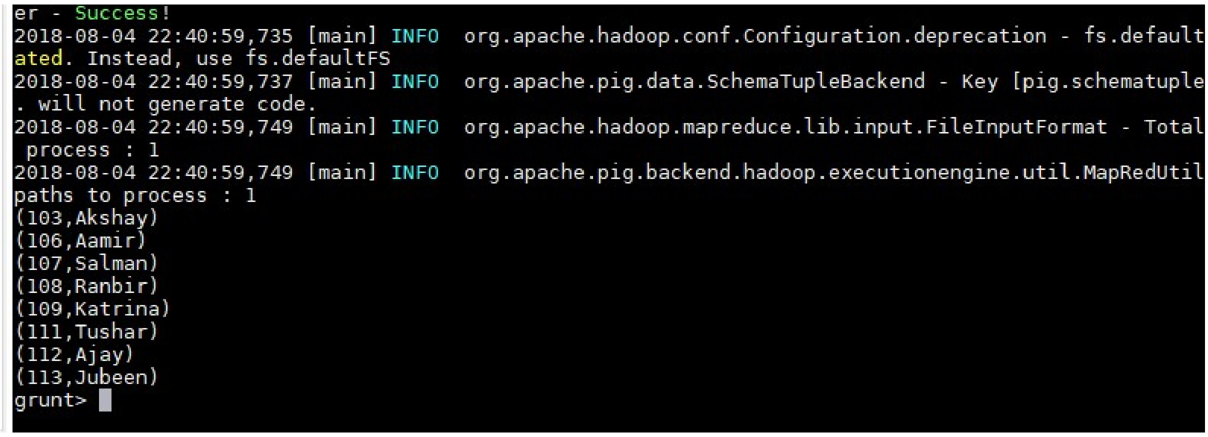
emp_expenses = LOAD '/hadoopdata/pig/employee_expenses.txt' AS (emp_id:int, expenses:int);

emp_without_exp = JOIN emp BY emp_id LEFT OUTER, emp_expenses BY emp_id;

emp_without_exp_filter = FILTER emp_without_exp BY emp_expenses::emp_id is null;

emp_without_exp_filter_data = FOREACH emp_without_exp_filter GENERATE emp::emp_id, emp::emp_name;

DUMP emp_without_exp_filter_data;



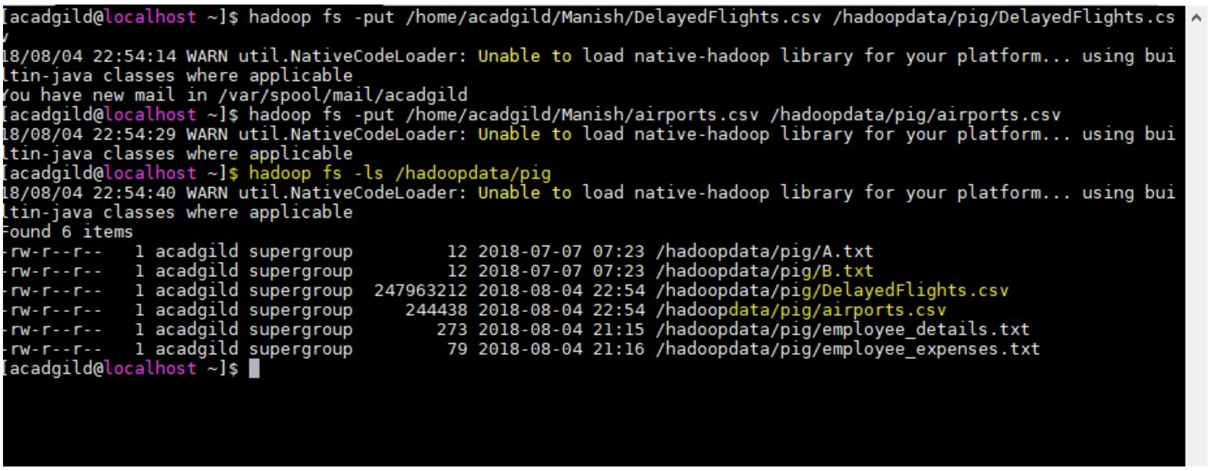
```
er - Success!
2018-08-04 22:40:59,735 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - fs.default
ated. Instead, use fs.defaultFS
2018-08-04 22:40:59,737 [main] INFO  org.apache.pig.data.SchemaTupleBackend - Key [pig.schematuple
. will not generate code.
2018-08-04 22:40:59,749 [main] INFO  org.apache.hadoop.mapreduce.lib.input.FileInputFormat - Total
 process : 1
2018-08-04 22:40:59,749 [main] INFO  org.apache.pig.backend.hadoop.executionengine.util.MapRedUtil
paths to process : 1
(103,Akshay)
(106,Aamir)
(107,Salman)
(108,Ranbir)
(109,Katrina)
(111,Tushar)
(112,Ajay)
(113,Jubeen)
grunt>
```

Task 3

Implement the use case present in below blog link and share the complete steps along with screenshot(s) from your end.

https://acadgild.com/blog/aviation-data-analysis-using-apache-pig/

Put the both the CSV file in HDFS



```
[acadgild@localhost ~]$ hadoop fs -put /home/acadgild/Manish/DelayedFlights.csv /hadoopdata/pig/DelayedFlights.cs
v
18/08/04 22:54:14 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using bui
ltin-java classes where applicable
You have new mail in /var/spool/mail/acadgild
[acadgild@localhost ~]$ hadoop fs -put /home/acadgild/Manish/airports.csv /hadoopdata/pig/airports.csv
18/08/04 22:54:29 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using bui
ltin-java classes where applicable
[acadgild@localhost ~]$ hadoop fs -ls /hadoopdata/pig
18/08/04 22:54:40 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using bui
ltin-java classes where applicable
Found 6 items
-rw-r--r--   1 acadgild supergroup         12 2018-07-07 07:23 /hadoopdata/pig/A.txt
-rw-r--r--   1 acadgild supergroup         12 2018-07-07 07:23 /hadoopdata/pig/B.txt
-rw-r--r--   1 acadgild supergroup  247963212 2018-08-04 22:54 /hadoopdata/pig/DelayedFlights.csv
-rw-r--r--   1 acadgild supergroup     244438 2018-08-04 22:54 /hadoopdata/pig/airports.csv
-rw-r--r--   1 acadgild supergroup        273 2018-08-04 21:15 /hadoopdata/pig/employee_details.txt
-rw-r--r--   1 acadgild supergroup         79 2018-08-04 21:16 /hadoopdata/pig/employee_expenses.txt
[acadgild@localhost ~]$
```

Problem Statement 1

Find out the top 5 most visited destinations.

REGISTER '/hadoopdata/pig/piggybank.jar';

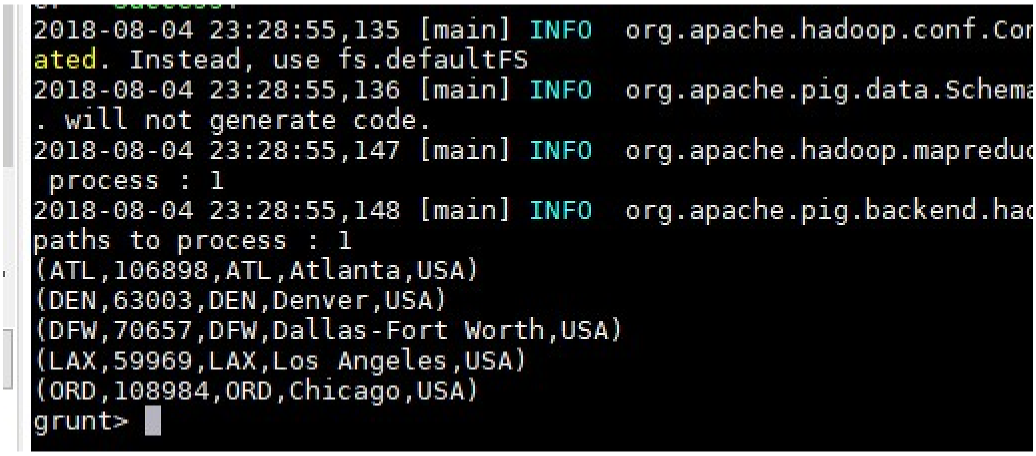
A = load '/hadoopdata/pig/DelayedFlights.csv' USING org.apache.pig.piggybank.storage.CSVExcelStorage(',','NO_MULTILINE','UNIX','SKIP_INPUT_HEADER');

B = foreach A generate (int)$1 as year, (int)$10 as flight_num, (chararray)$17 as origin,(chararray) $18 as dest;

C = filter B by dest is not null;

D = group C by dest;

```
E = foreach D generate group, COUNT(C.dest);

F = order E by $1 DESC;

Result = LIMIT F 5;

A1 = load 'airports.csv' USING org.apache.pig.piggybank.storage.CSVExcelStorage(',','NO_MULTILINE','UNIX','SKIP_INPUT_HEADER');

A2 = foreach A1 generate (chararray)$0 as dest, (chararray)$2 as city, (chararray)$4 as country;

joined_table = join Result by $0, A2 by dest;

dump joined_table;
```



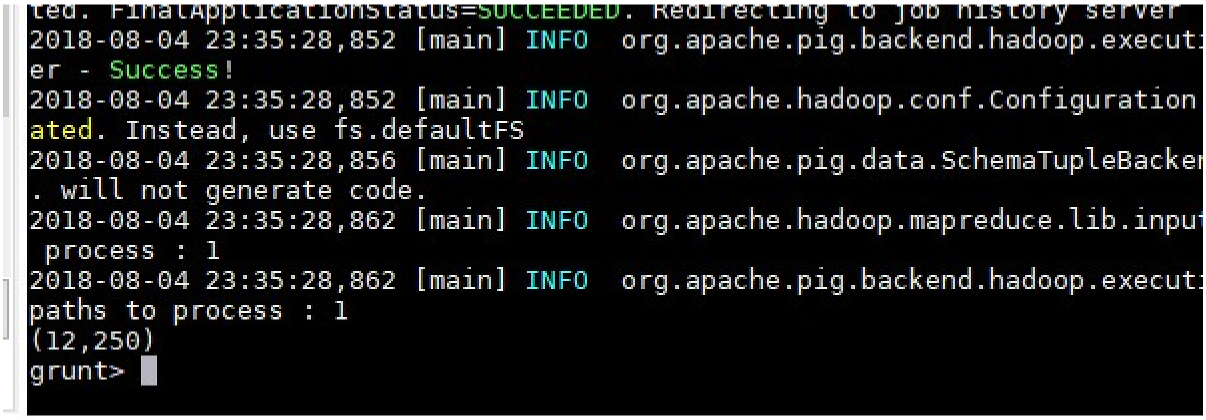
```
2018-08-04 23:28:55,135 [main] INFO  org.apache.hadoop.conf.Con
ated. Instead, use fs.defaultFS
2018-08-04 23:28:55,136 [main] INFO  org.apache.pig.data.Schema
. will not generate code.
2018-08-04 23:28:55,147 [main] INFO  org.apache.hadoop.mapreduc
 process : 1
2018-08-04 23:28:55,148 [main] INFO  org.apache.pig.backend.had
paths to process : 1
(ATL,106898,ATL,Atlanta,USA)
(DEN,63003,DEN,Denver,USA)
(DFW,70657,DFW,Dallas-Fort Worth,USA)
(LAX,59969,LAX,Los Angeles,USA)
(ORD,108984,ORD,Chicago,USA)
grunt>
```

Problem Statement 2

Which month has seen the most number of cancellations due to bad weather?

Source code

```
REGISTER '/hadoopdata/pig/piggybank.jar';

 A = load '/hadoopdata/pig/DelayedFlights.csv' USING org.apache.pig.piggybank.storage.CSVExcelStorage(',','NO_MULTILINE','UNIX','SKIP_INPUT_HEADER');

B = foreach A generate (int)$2 as month,(int)$10 as flight_num,(int)$22 as cancelled,(chararray)$23 as cancel_code;

C = filter B by cancelled == 1 AND cancel_code =='B';

D = group C by month;

E = foreach D generate group, COUNT(C.cancelled);

F= order E by $1 DESC;

Result = limit F 1;

dump Result;
```



```
ted. FinalApplicationStatus=SUCCEEDED. Redirecting to job history server
2018-08-04 23:35:28,852 [main] INFO  org.apache.pig.backend.hadoop.executi
er - Success!
2018-08-04 23:35:28,852 [main] INFO  org.apache.hadoop.conf.Configuration
ated. Instead, use fs.defaultFS
2018-08-04 23:35:28,856 [main] INFO  org.apache.pig.data.SchemaTupleBacken
. will not generate code.
2018-08-04 23:35:28,862 [main] INFO  org.apache.hadoop.mapreduce.lib.input
 process : 1
2018-08-04 23:35:28,862 [main] INFO  org.apache.pig.backend.hadoop.executi
paths to process : 1
(12,250)
grunt> 
```

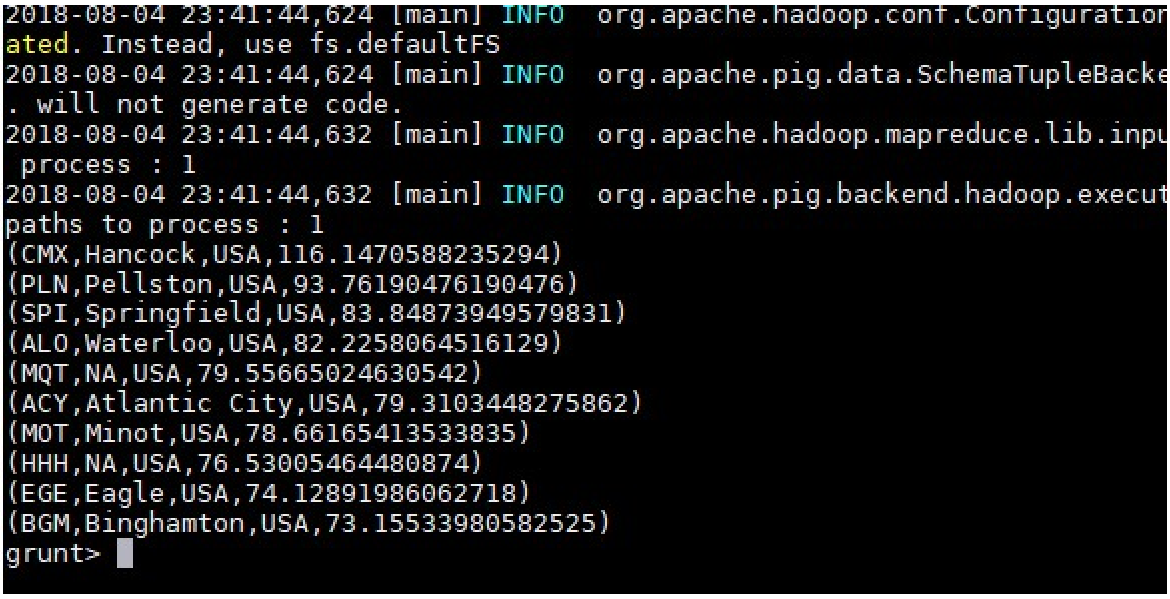
Problem Statement 3

Top ten origins with the highest AVG departure delay

Source code

```
A = load '/hadoopdata/pig/DelayedFlights.csv' USING org.apache.pig.piggybank.storage.CSVExcelStorage(',','NO_MULTILINE','UNIX','SKIP_INPUT_HEADER');

B1 = foreach A generate (int)$16 as dep_delay, (chararray)$17 as origin;

C1 = filter B1 by (dep_delay is not null) AND (origin is not null);

D1 = group C1 by origin;

E1 = foreach D1 generate group, AVG(C1.dep_delay);

Result = order E1 by $1 DESC;

Top_ten = limit Result 10;

Lookup = load '/hadoopdata/pig/airports.csv' USING org.apache.pig.piggybank.storage.CSVExcelStorage(',','NO_MULTILINE','UNIX','SKIP_INPUT_HEADER');

Lookup1 = foreach Lookup generate (chararray)$0 as origin, (chararray)$2 as city, (chararray)$4 as country;

Joined = join Lookup1 by origin, Top_ten by $0;

Final = foreach Joined generate $0,$1,$2,$4;

Final_Result = ORDER Final by $3 DESC;
```

dump Final_Result;



```
2018-08-04 23:41:44,624 [main] INFO  org.apache.hadoop.conf.Configuration
ated. Instead, use fs.defaultFS
2018-08-04 23:41:44,624 [main] INFO  org.apache.pig.data.SchemaTupleBacke
. will not generate code.
2018-08-04 23:41:44,632 [main] INFO  org.apache.hadoop.mapreduce.lib.inpu
 process : 1
2018-08-04 23:41:44,632 [main] INFO  org.apache.pig.backend.hadoop.execut
paths to process : 1
(CMX,Hancock,USA,116.1470588235294)
(PLN,Pellston,USA,93.76190476190476)
(SPI,Springfield,USA,83.84873949579831)
(ALO,Waterloo,USA,82.2258064516129)
(MQT,NA,USA,79.55665024630542)
(ACY,Atlantic City,USA,79.3103448275862)
(MOT,Minot,USA,78.66165413533835)
(HHH,NA,USA,76.53005464480874)
(EGE,Eagle,USA,74.12891986062718)
(BGM,Binghamton,USA,73.15533980582525)
grunt>
```

Problem Statement 4

Which route (origin & destination) has seen the maximum diversion?

Source code

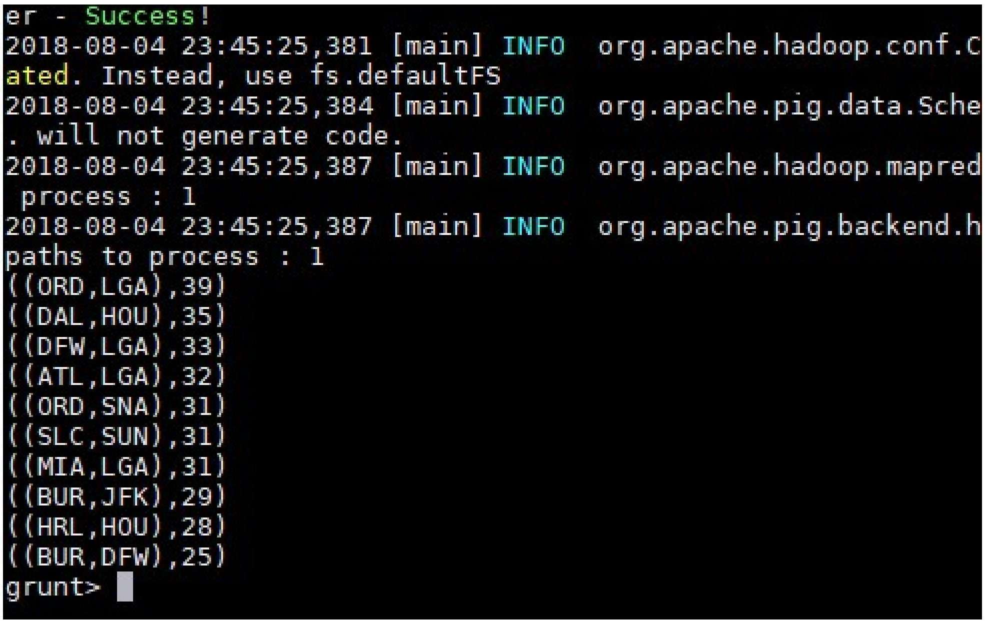
A = load '/hadoopdata/pig/DelayedFlights.csv' USING org.apache.pig.piggybank.storage.CSVExcelStorage(',','NO_MULTILINE','UNIX','SKIP_INPUT_HEADER');

B = FOREACH A GENERATE (chararray)$17 as origin, (chararray)$18 as dest, (int)$24 as diversion;

C = FILTER B BY (origin is not null) AND (dest is not null) AND (diversion == 1);

D = GROUP C by (origin,dest);

```
E = FOREACH D generate group, COUNT(C.diversion);

F = ORDER E BY $1 DESC;

Result = limit F 10;

dump Result;
```



```
er - Success!
2018-08-04 23:45:25,381 [main] INFO  org.apache.hadoop.conf.C
ated. Instead, use fs.defaultFS
2018-08-04 23:45:25,384 [main] INFO  org.apache.pig.data.Sche
. will not generate code.
2018-08-04 23:45:25,387 [main] INFO  org.apache.hadoop.mapred
 process : 1
2018-08-04 23:45:25,387 [main] INFO  org.apache.pig.backend.h
paths to process : 1
((ORD,LGA),39)
((DAL,HOU),35)
((DFW,LGA),33)
((ATL,LGA),32)
((ORD,SNA),31)
((SLC,SUN),31)
((MIA,LGA),31)
((BUR,JFK),29)
((HRL,HOU),28)
((BUR,DFW),25)
grunt> 
```