Task 1 Write a program to implement wordcount using Pig.

```
A = load '/tmp/alice.txt';

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

C = filter B by word matches '\\w+';

D = group C by word;

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

dump E

store E into '/tmp/alice_wordcount';
```

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

(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)

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

emp_rating = order emp by rating,emp_name;

dump emp_rating
```

(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)

```
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_sal by emp_salary desc;

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

emp_final = FOREACH emp_sal_id generate  emp_id,emp_name;

emp_final_limit = LIMIT emp_final 3;
dump emp_final_limit
```

(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)

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

Joinempempexpense = join emp by emp_id,empexpense by emp_id;
maxexpense = ORDER Joinempempexpense by empexpense::emp_expense desc;

Limitmaxepnse = LIMIT maxexpense 1;
Limitmaxexpensefinal = foreach Limitmaxepnse generate emp::emp_id,emp::emp_name;

dump Limitmaxexpensefinal;
```

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

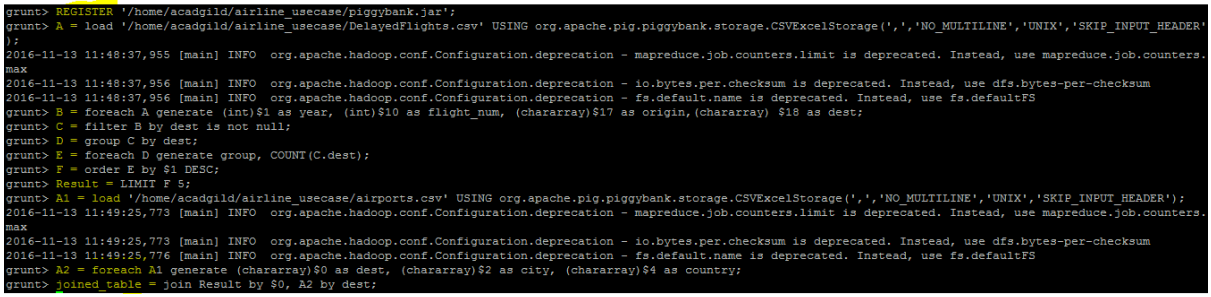
```
mp = LOAD 'employee_details' USING PigStorage(',') AS (emp_id:int, emp_name:chararray, emp_salary:int, emp_rating:int);

emp_expenses = LOAD 'employee_expenses' 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;
```

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

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

emp_expenses = LOAD '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;
```

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/

```
REGISTER '/home/acadgild/airline_usecase/piggybank.jar';
A = load '/home/acadgild/airline_usecase/DelayedFlights.csv' USING
org.apache.pig.piggybank.storage.CSVExcelStorage(',','NO_MULTILINE','UNIX','SKIP_INPUT_HEADE
R');
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 '/home/acadgild/airline_usecase/airports.csv' USING
org.apache.pig.piggybank.storage.CSVExcelStorage(',','NO_MULTILINE','UNIX','SKIP_INPUT_HEADE
R');
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;
```




```
REGISTER '/home/acadgild/airline_usecase/piggybank.jar';
A = load '/home/acadgild/airline_usecase/DelayedFlights.csv' USING
org.apache.pig.piggybank.storage.CSVExcelStorage(',','NO_MULTILINE','UNIX','SKIP_INPUT_HEADE
R');
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;
```

```
grunt> REGISTER '/home/acadgild/airline_usecase/piggybank.jar';
grunt> A = load '/home/acadgild/airline_usecase/DelayedFlights.csv' USING org.apache.pig.piggybank.storage.CSVExcelStorage(',','NO_MULTILINE','UNIX','SKIP_INPUT_HEADER'
);
2016-11-13 11:56:42,492 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - mapreduce.job.counters.limit is deprecated. Instead, use mapreduce.job.counters.
max
2016-11-13 11:56:42,493 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - io.bytes.per.checksum is deprecated. Instead, use dfs.bytes-per-checksum
2016-11-13 11:56:42,493 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - fs.default.name is deprecated. Instead, use fs.defaultFS
grunt> B = foreach A generate (int)$2 as month,(int)$10 as flight_num,(int)$22 as cancelled,(chararray)$23 as cancel_code;
grunt> C = filter B by cancelled == 1 AND cancel_code =='B';
grunt> D = group C by month;
grunt> E = foreach D generate group, COUNT(C.cancelled);
grunt> F= order E by $0 DESC;
grunt> Result = limit F 1;
```

```
2016-11-13 11:58:45,906 [main] INFO
max
2016-11-13 11:58:45,907 [main] WARN
2016-11-13 11:58:45,922 [main] INFO
2016-11-13 11:58:45,922 [main] INFO
(12,250)
```

REGISTER '/home/acadgild/airline_usecase/piggybank.jar';
A = load '/home/acadgild/airline_usecase/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 '/home/acadgild/airline_usecase/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;

```
grunt> REGISTER '/home/acadgild/airline_usecase/piggybank.jar';
grunt> A = load '/home/acadgild/airline_usecase/DelayedFlights.csv' USING org.apache.pig.piggybank.storage.CSVExcelStorage(',','NO_MULTILINE','UNIX','SKIP_INPUT_HEADER'
);
2016-11-13 12:20:54,240 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - mapreduce.job.counters.limit is deprecated. Instead, use mapreduce.job.counters.
max
2016-11-13 12:20:54,240 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - io.bytes.per.checksum is deprecated. Instead, use dfs.bytes-per-checksum
2016-11-13 12:20:54,240 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - fs.default.name is deprecated. Instead, use fs.defaultFS
grunt> B1 = foreach A generate (int)$16 as dep_delay, (chararray)$17 as origin;
grunt> C1 = filter B1 by (dep_delay is not null) AND (origin is not null);
grunt> D1 = group C1 by origin;
grunt> E1 = foreach D1 generate group, AVG(C1.dep_delay);
grunt> Result = order E1 by $1 DESC;
grunt> Top_ten = limit Result 10;
grunt> Lookup = load '/home/acadgild/airline_usecase/airports.csv' USING org.apache.pig.piggybank.storage.CSVExcelStorage(',','NO_MULTILINE','UNIX','SKIP_INPUT_HEADER')
;
2016-11-13 12:21:32,073 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - mapreduce.job.counters.limit is deprecated. Instead, use mapreduce.job.counters.
max
2016-11-13 12:21:32,074 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - io.bytes.per.checksum is deprecated. Instead, use dfs.bytes-per-checksum
2016-11-13 12:21:32,074 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - fs.default.name is deprecated. Instead, use fs.defaultFS
grunt> Lookup1 = foreach Lookup generate (chararray)$0 as origin, (chararray)$2 as city, (chararray)$4 as country;
grunt> Joined = join Lookup1 by origin, Top_ten by $0;
grunt> Final = foreach Joined generate $0,$1,$2,$4;
grunt> Final_Result = ORDER Final by $3 DESC;
```

```
(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)
```

REGISTER '/home/acadgild/airline_usecase/piggybank.jar';
A = load '/home/acadgild/airline_usecase/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;

```
grunt> REGISTER '/home/acadgild/airline_usecase/piggybank.jar';
grunt> A = load '/home/acadgild/airline_usecase/DelayedFlights.csv' USING org.apache.pig.piggybank.storage.CSVExcelStorage(',','NO_MULTILINE','UNIX','SKIP_INPUT_HEADER'
);
2016-11-13 12:29:00,015 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - mapreduce.job.counters.limit is deprecated. Instead, use mapreduce.job.counters.
max
2016-11-13 12:29:00,015 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - io.bytes.per.checksum is deprecated. Instead, use dfs.bytes-per-checksum
2016-11-13 12:29:00,015 [main] INFO  org.apache.hadoop.conf.Configuration.deprecation - fs.default.name is deprecated. Instead, use fs.defaultFS
grunt> B = FOREACH A GENERATE (chararray)$17 as origin, (chararray)$18 as dest, (int)$24 as diversion;
grunt> C = FILTER B BY (origin is not null) AND (dest is not null) AND (diversion == 1);
grunt> D = GROUP C by (origin,dest);
grunt> E = FOREACH D generate group, COUNT(C.diversion);
grunt> F = ORDER E BY $1 DESC;
grunt> Result = limit F 10;
```

```
(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)
```