Task 1(word count in pig)

Input file(word-count.txt)

```
[acadgild@localhost ~]$ cat /home/acadgild/word-count.txt
A wonderful king is Hadoop.
The elephant plays well with Sqoop.
But what helps him to thrive
Are Impala, and Hive,
And HDFS in the group.
Hadoop is an elegant fellow.
An elephant gentle and mellow.
He never gets mad,
Or does anything bad,
Because, at his core, he is yellow.
```

Executing pig script in local mode-

```
[acadgild@localhost ~]$ pig -x local /home/acadgild/Pig_Scripts/Pig_wordcount.pig
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/home/acadgild/install/hadoop/hadoop-2.6.5/share/h
mmon/lib/slf4j-log4j12-1.7.5.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/home/acadgild/install/hbase/hbase-1.2.6/lib/slf4j
-1.7.5.jar!/org/slf4j/impl/StaticLoggerBinder.class]
```

Output-

```
2018-10-24 22:53:43,787 [main] INFO  org.apa
edUtil - Total input paths to process : 1
(A,1)
(An,1)
(He,1)
(Or,1)
(an,1)
(at,1)
(he,1)
(in,1)
(is,3)
(to,1)
(And,1)
(Are,1)
(But,1)
(The,1)
(and,2)
(bad,1)
(him,1)
(his,1)
(mad,1)
(the,1)
(HDFS,1)
(Hive,1)
```

Pig Scripts-

```
data_set = load '/home/acadgild/word-count.txt' using PigStorage('\t') as (textdata:chararray);
text_row = foreach data_set generate FLATTEN((TOKENIZE(textdata)))as word;
group_text_row = group text_row by word;
word_count = foreach group_text_row generate group as word,COUNT(text_row) as counts;
dump word_count;
```

Task 2

---

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)

Pig Script-

```
emp_dataset = load '/home/acadgild/employee_details.txt' using PigStorage(',') as
(EmpID:int,Name:chararray,Salary:long,EmployeeRating:int);
emp_orderby = ORDER emp_dataset BY EmployeeRating ASC,Name ASC;
emp_limit = LIMIT emp_orderby 5;
emp_result = FOREACH emp_limit GENERATE EmpID as EmployeeID,Name as
EmployeeName;
dump emp_result;
```

Output:

```
il - Total input paths to process : 1
(106,Aamir)
(101,Amitabh)
(113,Jubeen)
(111,Tushar)
(112,Ajay)
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 Script-

```
emp_dataset = load '/home/acadgild/employee_details.txt' using PigStorage(',') as
(EmpID:int,Name:chararray,Salary:long,EmployeeRating:int);
emp_dataset_odd= FILTER emp_dataset BY EmpID%2==1;
emp_orderby = ORDER emp_dataset_odd BY Salary DESC,Name ASC;
emp_limit = LIMIT emp_orderby 3;
emp_result = FOREACH emp_limit GENERATE EmpID as EmployeeID,Name as
EmployeeName;
dump emp_result;
```

Output:

```
(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 Script-

```
emp_dataset = load '/home/acadgild/employee_details.txt' using PigStorage(',') as
(EmpID:int,Name:chararray,Salary:long,EmployeeRating:int);
emp_exp_dataset =load '/home/acadgild/employee_expenses.txt' using PigStorage('\t') as
(EmpID:int,Expense:int);
emp_join = JOIN emp_dataset BY EmpID,emp_exp_dataset BY EmpID;
emp_exp_order = ORDER emp_join BY Expense DESC,emp_dataset::Name ASC;
emp_exp_max = LIMIT emp_exp_order 1;
emp_res = FOREACH emp_exp_max GENERATE emp_dataset::EmpID,emp_dataset::Name;
dump emp_res;
```

Output:

```
2018-10-25 01:22:56,137 [mai
] input paths to process : 1
(110,Priyanka)
```

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

Pig Script-

```
emp_dataset = load '/home/acadgild/employee_details.txt' using PigStorage(',') as
(EmpID:int,Name:chararray,Salary:long,EmployeeRating:int);
emp_exp_dataset =load '/home/acadgild/employee_expenses.txt' using PigStorage('\t') as
(EmpID:int,Expense:int);
emp_join = JOIN emp_dataset BY EmpID,emp_exp_dataset BY EmpID;
emp_res = FOREACH emp_join GENERATE emp_dataset::EmpID,emp_dataset::Name;
```

```
emp_dis = DISTINCT emp_res;
DUMP emp_dis;
```

Output:

```
(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 Script-

```
emp_dataset = load '/home/acadgild/employee_details.txt' using PigStorage(',') as
(EmpID:int,Name:chararray,Salary:long,EmployeeRating:int);
emp_exp_dataset =load '/home/acadgild/employee_expenses.txt' using PigStorage('\t') as
(EmpID:int,Expense:int);
emp_join = JOIN emp_dataset BY EmpID LEFT OUTER,emp_exp_dataset BY EmpID;
emp_filter = FILTER emp_join BY emp_exp_dataset::EmpID is null;
emp_res = FOREACH emp_filter GENERATE emp_dataset::EmpID,emp_dataset::Name;
emp_dis = DISTINCT emp_res;
DUMP emp_dis;
```

Output:

```
2018-10-25 01:42:3
l input paths to p
(103,Akshay)
(106,Aamir)
(107,Salman)
(108,Ranbir)
(109,Katrina)
(111,Tushar)
(112,Ajay)
(113,Jubeen)
grunt>
```

Task 3(Aviation Use Case)

---

a) **Find out the top 5 most visited destinations**

Code –

```
REGISTER '/home/acadgild/Downloads/piggybank-0.15.0.jar';
A = load '/home/acadgild/Downloads/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;
airport = load '/home/acadgild/Downloads/airports.csv' USING
org.apache.pig.piggybank.storage.CSVExcelStorage(',','NO_MULTILINE','UNIX','SKIP_INPUT_HEADER')
;
A2 = foreach airport generate (chararray)$0 as dest, (chararray)$2 as city, (chararray)$4 as country;
final_res = join Result by $0, A2 by dest;
dump final_res;
```

Output-

```
2018-12-17 22:36:00,319 [main] INFO  org.apache
paths to process : 1
2018-12-17 22:36:00,319 [main] INFO  org.apache
] input 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>
```

b) Which month has seen the most number of cancellations due to bad weather.

Code-

```
REGISTER '/home/acadgild/Downloads/piggybank-0.15.0.jar';
DF = load '/home/acadgild/Downloads/DelayedFlights.csv' USING
org.apache.pig.piggybank.storage.CSVExcelStorage(',','NO_MULTILINE','UNIX','SKIP_INPUT_HEADER')
;
B = foreach DF 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_cancel = limit F 1;
dump Result_cancel;
```

Output-

```
paths to process : 1
2018-12-17 22:45:56,787 [main
l input paths to process : 1
(12,250)
grunt>
```

c) Top ten origins with the highest AVG departure delay.

Code –

```
REGISTER '/home/acadgild/Downloads/piggybank-0.15.0.jar';
A = load '/home/acadgild/Downloads/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/Downloads/airports.csv' USING
org.apache.pig.piggybank.storage.CSVExcelStorage(',','NO_MULTILINE','UNIX','SKIP_INPUT_HEADER')
;
Lookup_country = foreach Lookup generate (chararray)$0 as origin, (chararray)$2 as city,
(chararray)$4 as country;
Joined = join Lookup_country by origin, Top_ten by $0;
Final = foreach Joined generate $0,$1,$2,$4;
Final_Result = ORDER Final by $3 DESC;
dump Final_Result;
```

Output-

```
2018-12-17 23:19:50,546 [main] INFO  org.a
paths to process : 1
2018-12-17 23:19:50,546 [main] INFO  org.a
l input 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>
```

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

Code-

```
REGISTER '/home/acadgild/Downloads/piggybank-0.15.0.jar';
flights = load '/home/acadgild/Downloads/DelayedFlights.csv' USING
org.apache.pig.piggybank.storage.CSVExcelStorage(',','NO_MULTILINE','UNIX','SKIP_INPUT_HEADER')
;
flights_set = FOREACH flights GENERATE (chararray)$17 as origin, (chararray)$18 as dest, (int)$24 as
diversion;
diversion_set = FILTER flights_set BY (origin is not null) AND (dest is not null) AND (diversion == 1);
D = GROUP diversion_set by (origin,dest);
E = FOREACH D generate group, COUNT(diversion_set.diversion);
F = ORDER E BY $1 DESC;
Result = limit F 2;
dump Result;
```

Output- showing top 2 destination with highest diversion.