LAB- 7

AIM: To implement PIG commands.

COMMANDS/ QUERIES:

Command 1: emp = LOAD '/home/cloudera/Desktop/random.txt' USING PigStorage(',') as (id:int, name:chararrary, date_str:chararray, salary:int, bonus:int, city:chararray)

```
Input(s):
Successfully read records from: "/home/cloudera/Desktop/random.txt"

Output(s):
Successfully stored records in: "file:/tmp/temp191057885/tmp1298460806"

Job DAG:
job_local1151205579_0001        ->      job_local279820316_0002,
job_local279820316_0002 ->      job_local1849206003_0003,
job_local1849206003_0003        ->      job_local1321328117_0004,
job_local1321328117_0004
```

Command 2: emp_order = ORDER emp by salary DESC;

```
2023-05-01 12:09:59,393 [main] INFO  org.apache.hadoop.mapreduce.lib.input.FileI
nputFormat - Total input paths to process : 1
2023-05-01 12:09:59,393 [main] INFO  org.apache.pig.backend.hadoop.executionengi
ne.util.MapRedUtil - Total input paths to process : 1
(3,Arpit,2018-01-06,50000,1150,Chennai)
(5,Trisha,2016-12-31,46500,1530,Hyderabad )
(1,Satish,2016-04-23,42000,1250,Bhopal)
(4,Tanya,2013-10-03,39000,1800,Delhi)
grunt> clear
```

Command 3: emp_limit = LIMIT emp_order 4;

```
2023-05-01 12:09:59,393 [main] INFO  org.apache.hadoop.mapreduce.lib.input.FileI
nputFormat - Total input paths to process : 1
2023-05-01 12:09:59,393 [main] INFO  org.apache.pig.backend.hadoop.executionengi
ne.util.MapRedUtil - Total input paths to process : 1
(3,Arpit,2018-01-06,50000,1150,Chennai)
(5,Trisha,2016-12-31,46500,1530,Hyderabad )
(1,Satish,2016-04-23,42000,1250,Bhopal)
(4,Tanya,2013-10-03,39000,1800,Delhi)
grunt> clear
```

Command 4: DUMP emp_limit;

```
grunt> DUMP emp_limit;
```

Command 5: emp_filter = FILTER emp BY salary > 42000;

```
l.MapRedUtil - Total input paths to process : 1
(3,Arpit,2018-01-06,50000,1150,Chennai)
(5,Trisha,2016-12-31,46500,1530,Hyderabad )
grunt> clear
```

Command 6: emp_xtra = LOAD '/home/cloudera/Desktop/randomdat.txt' USING PigStorage(',') as joined_data = JOIN emp BY id, emp_xtra BY xid;

```
l.MapRedUtil - Total input paths to process : 1
(1,Satish,2016-04-23,42000,1250,Bhopal,1,B plus,5769120452)
(2,Dinesh,2017-06-13,34000,1650,Allhabad,2,O plus,9800231452)
(3,Arpit,2018-01-06,50000,1150,Chennai,3,B plus,8351849149)
(4,Tanya,2013-10-03,39000,1800,Delhi,4,AB plus,7899943141)
(5,Trisha,2016-12-31,46500,1530,Hyderabad ,5,A minus,1735613856)
grunt> clear
```

Command 7: emp_group = GROUP emp_filter BY id;
Emp_trans = FOREACH emp_group GENERATE group, SUM(emp_filter.salary)

```
  rmat - Total input paths to process : 1
  2023-05-01 13:28:25,594 [main] INFO  org.apache.pig.backend.hadoop.executionengine.uti
  l.MapRedUtil - Total input paths to process : 1
  (3,50000)
  (5,46500)
```

Command 8: sampled_data = SAMPLE emp 0.1;

```
l.MapRedUtil - Total input paths to process : 1
(2,Dinesh,2017-06-13,34000,1650,Allhabad)
grunt> clear
```

Command 9: SPLIT emp INTO emp_young IF age < 30, emp_elder OTHERWISE;

```
2023-05-20 21:12:21,421 [main] INFO  org.apache.pig.backend.hadoop.executionengine.util.MapRedUtil - Total input paths to pr
ocess : 1
(1001,raju,24,45000,R&D)
(1002,shyam,23,55000,R&D)
(1006,anjali,28,63000,sales)
(2021,anuj,28,53000,marketing)
(2022,anjana,27,53000,marketing)
(2023,sita,26,52000,marketing)
grunt>
```

```
2023-05-20 21:14:15,280 [main] INFO  org.apache.pig.backend.hadoop.executionengine.util.MapRedUtil - Total input paths to pr
ocess : 1
(1003,rao,42,72000,sales)
(1004,meera,32,64000,sales)
(1005,neha,33,65000,sales)
(2034,ram,35,65000,HR)
(2035,ritu,35,65000,HR)
(20364,raghu,30,65000,HR)
(2026,geeta,37,52000,marketing)
grunt>
```

Command 10: word_count = LOAD ‘/home/cloudera/Desktop/sample.txt’ AS (lines:chararray);
Words = FOREACH word_count GENERATE FLATTEN (TOKENIZE(lines, ‘’)) AS word;
Word_group = GROUP words BY word;
Count_word = FOREACH word_group GENERATE group, count(words);

```
2023-05-20 21:54:47,508 [main] INFO  org.apache.pig.backend.hadoop.executionengine.util.Map
RedUtil - Total input paths to process : 1
(a,5)
(in,1)
(is,2)
(of,3)
(to,1)
(Pig,2)
(The,1)
(and,1)
(bag,1)
(for,1)
(the,2)
(This,1)
(file,1)
(used,1)
(word,1)
(Latin,1)
(count,1)
(group,1)
(split,2)
(tuple,1)
(using,1)
(which,1)
((which,1)
(Latin.,1)
(output,1)
(sample,1)
(single,1)
(string,1)
(words),1)
(program,1)
(returns,1)
(running,1)
(contains,2)
(function,1)
```