**Q1 Script**

airline\_data = load '/user/pig/airlines.dat' using PigStorage(',') AS (id:int, name:chararray, alias:chararray, iata: chararray, icao:chararray, callsign:chararray, country:chararray, active:chararray);

airCanName = filter airline\_data by (name matches '.\*air canada\*.' or name matches '.\*Air Canada\*.' or name matches '.\*AirCanada\*' or name matches '.\*aircanada\*.' or name matches '.\*AIRCANADA\*.');

airCanName\_filtered = foreach airCanName generate(id, name);

dump airCanName\_filtered;

**Output**



**Q2 script**

airport\_data = load '/user/pig/airports.dat' using PigStorage (',') AS (id:int, name:chararray, city:chararray, country:chararray, iata:chararray, icao:chararray, lat:float, lon:float, altitude:int, timezone:float, dst:chararray, tz:chararray, type:chararray, source:chararray);

airportCountry = group airport\_data by country;

airportCount = foreach airportCountry generate group as country, COUNT(airport\_data.name) as cnt;

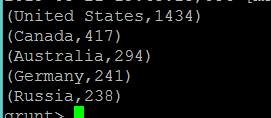
result = foreach airportCount generate country, cnt;

orderedResult = order result by cnt desc;

limitResult = limit orderedResult 5;

dump limitResult;

**Output**



**Q3 script**

route\_data = load ‘/user/pig/routes.dat’ using PigStorage(‘,’) as

(airline:chararray, id:int, source:chararray, sourceId:int, dest:chararray, destId, int, codeshare:chararray, stops:int, equipment:chararray);

route\_data\_limit = limit route\_data 5

groupRoute = group route\_data limit all;

distRoute = foreach groupRoute {

a=route\_data\_limit.(airline,source, dest);

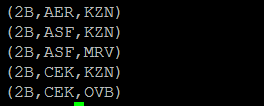
b=distinct a;

generate FLATTEN(b);

}

dump distRoute;

**Output**



**Question 4 script**

route\_data = load ‘/user/pig/routes.dat’ using PigStorage(‘,’) as

(airline:chararray, id:int, source:chararray, sourceId:int, dest:chararray, destId:int, codeshare:chararray, stops:int, equipment:chararray);

airport\_data = load ‘/user/pig/airports.dat’ using PigStorage (‘,’) as (id:int, name:chararray, city:chararray, country:chararray, iata:chararray, icao:chararray, lat:float, lon:float, altitude:int, timezone:float, dst:chararray, tz:chararray, type:chararray, source:chararray);

q4table\_source = join airport\_data by $0, route\_data by $3

q4table\_dest = join airport\_data by $0, route\_data by $5;

q4table\_sourcefinal = foreach q4table\_source generate airport\_data::id as sourceAirport\_id, airport\_data::name as sourceAirport\_name, airport\_data::lat as sourceAirport\_lat, airport\_data::lon as sourceAirport\_lon, route\_data::id as sourceRoute\_id;

q4table\_destfinal = foreach q4table\_dest generate airport\_data::id as destAirport\_id, airport\_data::name as destAirport\_name, airport\_data::lat as destAirport\_lat, airport\_data::lon as destAirport\_lon, route\_data::id as destRoute\_id;

q4table\_sourcefinalDist = distinct q4table\_sourcefinal;

q4table\_destfinalDist = distinct q4table\_destfinal;

q4table\_final = join q4table\_sourcefinalDist by $4, q4table\_destfinalDist by $4;

distances = foreach q4table\_final generate $0, $1, $5, $6, SQRT((destAirport\_lat - sourceAirport\_lat) \* (destAirport\_lat - sourceAirport\_lat) + (destAirport\_lon - sourceAirport\_lon) \* (destAirport\_lon - sourceAirport\_lon)) \* 111;

store distances into ‘/user/pig/routes\_with\_distances’ using PigStorage(‘\t’);

ls /user/pig/routes\_with\_distances/ ;



\*\*Outside of grunt, using local shell:

hadoop fs -cat /user/pig/routes\_with\_distances/part-r-0000 | head -n 5;

**Output**

