CSP 554 – Assignment #5

Name: Adarsh Mathad Vijayakumar
CWID: A20424847
Email ID: avijayakumar@hawk.iit.edu

Exercise 1)

Create new versions of the foodratings and foodplaces files by using TestDataGen (as described in assignment #4) and copy them to HDFS

Command: java TestDataGen

Output: Magic Number = 37931

Write and execute a sequence of pig latin statements that loads the foodratings file as a relation. Call the relation 'food_ratings'.

Commands:

1. food_ratings = LOAD '/user/maria_dev/foodratings37931.txt' USING PigStorage(',') AS (name:chararray, f1:int, f2:int, f3:int, f4:int, placeid:int);
2. DESCRIBE food_ratings;

```
grunt> food_ratings = LOAD '/user/maria_dev/ foodratings37931.txt' USING PigStorage(',') AS (name:chararray,
f1:int, f2:int, f3:int, f4:int, placeid:int);
grunt>
grunt> DESCRIBE food_ratings;
food_ratings: {name: chararray,f1: int,f2: int,f3: int,f4: int,placeid: int}
grunt>
```

Exercise 2)

Now create another relation with two fields of the initial (food_ratings) relation: 'name' and 'f4'. Call this relation 'food_ratings_subset'.

Store this last relation back to HDFS.

Also write 6 records of this relation out to the console.

Commands:

1. food_ratings_subset = FOREACH food_ratings GENERATE name, f4;
2. STORE food_ratings_subset INTO '/user/maria_dev/food_ratings_subset' USING PigStorage(',');
3. Top6_food_ratings_subset = LIMIT food_ratings_subset 6;
4. DUMP Top6_food_ratings_subset;

```
7353/tmp-1861824495/_temporary/0/task__0001_m_000001
2019-03-10 19:40:57,632 [main] WARN  org.apache.pig.data.SchemaTupleBackend - SchemaTupleBackend has already been initialized
2019-03-10 19:40:57,637 [main] INFO  org.apache.hadoop.mapreduce.lib.input.FileInputFormat - Total input paths to process : 1
2019-03-10 19:40:57,637 [main] INFO  org.apache.pig.backend.hadoop.executionengine.util.MapRedUtil - Total input paths to process : 1
(Joy,30)
(Sam,26)
(Mel,4)
(Jill,38)
(Joy,34)
(Jill,13)
grunt> _
```

Exercise 3)

Now create another relation using the initial (food_ratings) relation. Call this relation 'food_ratings_profile'. The new relation should only have one record. This record should hold the minimum, maximum and average values for the attributes 'f2' and 'f3'. (So this one record will have 6 fileds).

Commands:

1. food_ratings_profile = FOREACH (GROUP food_ratings ALL) GENERATE MIN(food_ratings.f2), MAX(food_ratings.f2), AVG(food_ratings.f2), MIN(food_ratings.f3), MAX(food_ratings.f3), AVG(food_ratings.f3);

2. DUMP food_ratings_profile;

```
Input(s):
Successfully read 1000 records (17504 bytes) from: "/user/maria_dev/foodratings37931.txt"

Output(s):
Successfully stored 1 records (28 bytes) in: "hdfs://sandbox-hdp.hortonworks.com:8020/tmp/temp-1320387353/tmp341044615"

2019-03-10 19:51:26,728 [main] INFO  org.apache.hadoop.mapreduce.lib.input.FileInputFormat - Total input paths to process : 1
2019-03-10 19:51:26,728 [main] INFO  org.apache.pig.backend.hadoop.executionengine.util.MapRedUtil - Total input paths to process : 1
(1,50,25.905,1,50,25.098)
grunt>
```

Exercise 4)

Now create yet another relation from the initial (food_ratings) relation. This new relation should only include tuples (records) where f1 < 20 and f3 > 5. Call this relation 'food_ratings_filtered'.

Write 6 records of this relation out to the console.

Commands:

1. food_ratings_filtered = FILTER food_ratings BY (f1 < 20) AND (f3 > 5);
2. Top6_food_ratings_filtered = LIMIT food_ratings_filtered 6;
3. DUMP Top6_food_ratings_filtered;

```
Input(s):
Successfully read 15 records (17504 bytes) from: "/user/maria_dev/foodratings37931.txt"

Output(s):
Successfully stored 6 records (122 bytes) in: "hdfs://sandbox-hdp.hortonworks.com:8020/tmp/temp-1320387353/tmp-1985776036"

2019-03-10 19:59:40,135 [main] INFO  org.apache.hadoop.mapreduce.lib.input.FileInputFormat - Total input paths to process : 1
2019-03-10 19:59:40,135 [main] INFO  org.apache.pig.backend.hadoop.executionengine.util.MapRedUtil - Total input paths to process : 1
(Jill,13,50,48,38,5)
(Joy,5,26,35,34,3)
(Jill,2,43,6,13,2)
(Mel,18,4,8,29,1)
(Joy,6,11,19,34,3)
(Jill,15,5,22,35,3)
grunt>
```

Exercise 5)

Using the initial (food_ratings) relation, write and execute a sequence of pig latin statements that creates another relation, call it 'food_ratings_2percent', holding a random selection of 2% of the records in the initial relation.

Write 10 of the records out to the console.

Commands:

1. food_ratings_2percent = SAMPLE food_ratings 0.02;
2. Top10_food_ratings_2percent = LIMIT food_ratings_2percent 10;
3. DUMP Top10_food_ratings_2percent;

```
Input(s):
Successfully read 223 records (17504 bytes) from: "/user/maria_dev/foodratings37931.txt"

Output(s):
Successfully stored 10 records (199 bytes) in: "hdfs://sandbox-hdp.hortonworks.com:8020/tmp/temp-1320387353/tmp-2146242747"

2019-03-10 20:03:39,182 [main] INFO  org.apache.hadoop.mapreduce.lib.input.FileInputFormat - Total input paths to process : 1
2019-03-10 20:03:39,183 [main] INFO  org.apache.pig.backend.hadoop.executionengine.util.MapRedUtil - Total input paths to process : 1
(Joy,6,11,19,34,3)
(Mel,18,1,42,12,5)
(Jill,31,37,5,36,1)
(Sam,42,45,5,34,1)
(Jill,8,24,39,14,1)
(Mel,45,22,22,19,3)
(Jill,30,2,10,15,5)
(Jill,49,42,11,33,1)
(Sam,10,8,16,34,3)
(Sam,11,22,47,15,2)
grunt>
```

Exercise 6)

Write and execute a sequence of pig latin statements that loads the foodplaces file as a relation. Call the relation 'food_places'.

Execute the describe command on this relation.

Commands:

1. food_places = LOAD '/user/maria_dev/foodplaces37931.txt' USING PigStorage(',') AS (placeid:int, placename:chararray);
2. DESCRIBE food_places;

```
grunt> food_places = LOAD '/user/maria_dev/foodplaces37931.txt' USING PigStorage(',') AS (placeid:int, placename:chararray);
grunt> DESCRIBE food_places;
food_places: {placeid: int,placename: chararray}
grunt>
```

Now perform a join between the initial food_ratings relation and the food_places relation on the placeid attributes to create a new relation called 'food_ratings_w_place_names'.

Write 6 records of this relation out to the console.

Commands:

1. food_ratings_w_place_names = JOIN food_ratings BY placeid, food_places BY placeid;
2. Top6_food_ratings_w_place_names = LIMIT food_ratings_w_place_names 6;
3. DUMP Top6_food_ratings_w_place_names;

```
Input(s):
Successfully read 5 records (59 bytes) from: "/user/maria_dev/foodplaces37931.txt"
Successfully read 1000 records (17504 bytes) from: "/user/maria_dev/foodratings37931.txt"

Output(s):
Successfully stored 6 records (211 bytes) in: "hdfs://sandbox-hdp.hortonworks.com:8020/tmp/temp-1320387353/tmp1509587446"

2019-03-10 20:13:34,248 [main] INFO  org.apache.hadoop.mapreduce.lib.input.FileInputFormat - Total input paths to process : 1
2019-03-10 20:13:34,248 [main] INFO  org.apache.pig.backend.hadoop.executionengine.util.MapRedUtil - Total input paths to process : 1
(Mel,34,20,33,34,1,1,China Bistro)
(Joe,23,46,28,40,1,1,China Bistro)
(Mel,18,4,8,29,1,1,China Bistro)
(Jill,30,43,27,34,1,1,China Bistro)
(Joe,45,32,2,31,1,1,China Bistro)
(Joe,2,20,27,10,1,1,China Bistro)
grunt>
```

7) Which keyword is used to select a certain number of rows from a relation when forming a new relation?

Answer: **Option A. LIMIT**

Choices:
A. LIMIT
B. DISTINCT
C. UNIQUE
D. SAMPLE

8) Which keyword returns only unique rows for a relation when forming a new relation?
Answer: C. Distinct
A. SAMPLE
B. FILTER
C. DISTINCT
D. SPLIT

9) Assume you have an HDFS file with a large number of records similar to the examples below
   - Mel, 1, 2, 3
   - Jill, 3, 4, 5

   Which of the following would NOT be a correct pig schema for such a file?

   Answer: B. (f1: STRING, f2: INT, f3: INT, f4: INT)

A. (f1: CHARARRY, f2: INT, f3: INT, f4: INT)
B. (f1: STRING, f2: INT, f3: INT, f4: INT)
C. (f1, f2, f3, f4)
D. (f1: BYTEARRAY, f2: INT, f3: BYTEARRAY, f4: INT)

10) Which one of the following statements would create a relation (relB) with two columns from a relation (relA) with 4 columns? Assume the pig schema for relA is as follows:
(f1: INT, f2, f3, f4: FLOAT)

Answer: option B
Choices:
A. relB = GROUP relA GENERATE f1, f3;
B. relB = FOREACH relA GENERATE $0, f3;
C. relB = FOREACH relA GENERATE f1, f5;
D. relB = FOREACH relA SELECT f1, f3;

11) Pig Latin is a _____ language. Select the best choice to fill in the blank.

Answer: Option B – Data Flow

Choices:

A. functional
B. data flow
C. procedural
D. declarative

12) Given a relation (relA) with 4 columns and pig schema as follows: (f1: INT, f2, f3, f4: FLOAT) which one statement will create a relation (relB) having records all of whose first field is less than 20

Answer: Option A

Choices:

A. relB = FILTER relA by $0 < 20
B. relB = GROUP relA by f1 < 20
C. relB = FILTER relA by $1 < 20
D. relB = FOREACH relA GENERATE f1 < 20