Chirag Khandhar

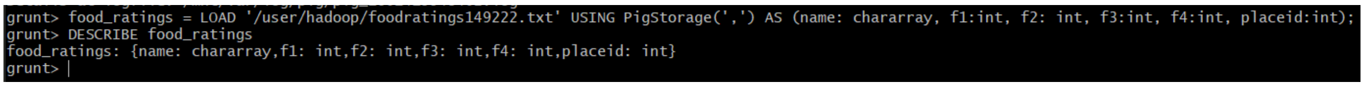
A20438926 | CSP 554 - Big Data Technologies | Fall 2020 | Assignment 5

Magic Number = **149222**

Q 1. Write and execute a sequence of pig latin statements that loads the foodratings file as a relation. Call the relation 'food_ratings'. The load command should associate a schema with this relation where the first attribute is referred to as 'name' and is of type chararray, the next attributes are referred to as 'f1' through 'f4' and are of type int, and the last field is referred to as 'placeid' and is also of type int.

```
food_ratings = LOAD '/user/hadoop/foodratings149222.txt' USING PigStorage(',') AS (name:
chararray, f1:int, f2: int, f3:int, f4:int, placeid:int);

DESCRIBE food_ratings;
```



```
grunt> food_ratings = LOAD '/user/hadoop/foodratings149222.txt' USING PigStorage(',') AS (name: chararray, f1:int, f2: int, f3:int, f4:int, placeid:int);
grunt> DESCRIBE food_ratings
food_ratings: {name: chararray,f1: int,f2: int,f3: int,f4: int,placeid: int}
grunt> |
```

Q 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, food_ratings_subset, back to HDFS (perhaps as the file /user/hadoop/fr_subset)

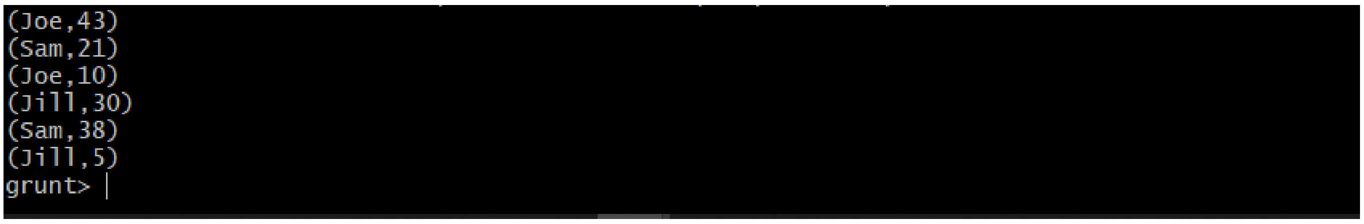
Also write 6 records of this relation out to the console./

```
food_ratings_subset = FOREACH food_ratings GENERATE name, f4;

STORE food_ratings_subset INTO '/user/hadoop/fr_subset ' USING PigStorage(',');

Result = LIMIT food_ratings_subset 6;

DUMP Result;
```



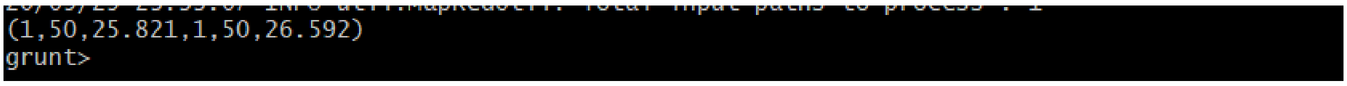
```
(Joe,43)
(Sam,21)
(Joe,10)
(Jill,30)
(Sam,38)
(Jill,5)
grunt> |
```

Q 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).
Write the record of this relation out to the console.

```
fr_relation = GROUP food_ratings ALL;

food_ratings_profile = FOREACH fr_relation 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);

DUMP food_ratings_profile;
```



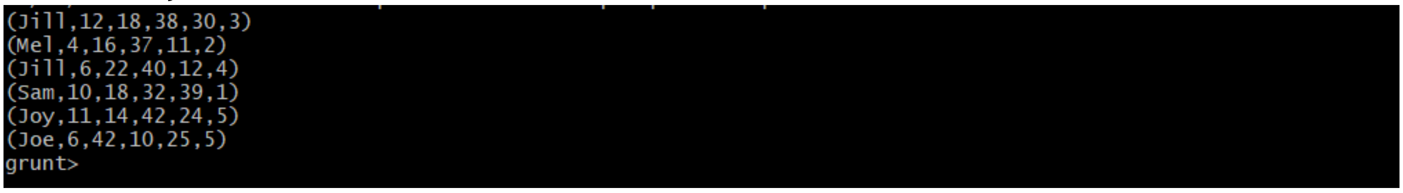
```
(1,50,25.821,1,50,26.592)
grunt>
```

Q 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.

```
food_ratings_filtered = FILTER food_ratings BY (f1 < 20) AND (f3 > 5);

Result = LIMIT food_ratings_filtered 6;

DUMP Result;
```



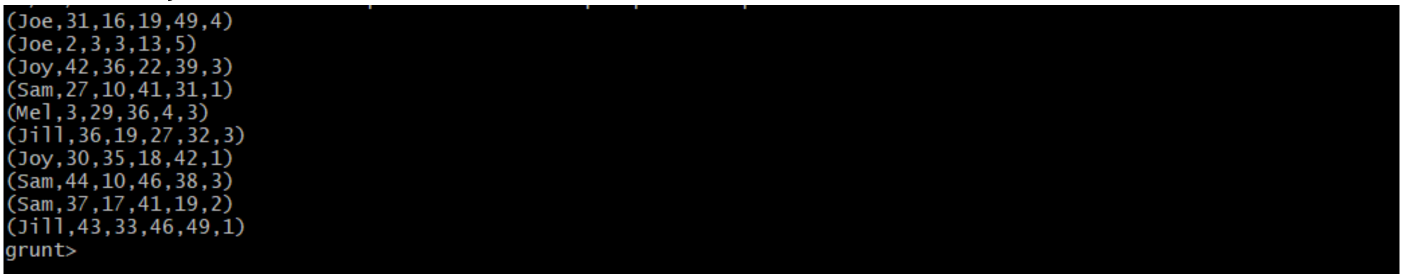
```
(Jill,12,18,38,30,3)
(Mel,4,16,37,11,2)
(Jill,6,22,40,12,4)
(Sam,10,18,32,39,1)
(Joy,11,14,42,24,5)
(Joe,6,42,10,25,5)
grunt>
```

Q 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.

```
food_ratings_2percent = SAMPLE food_ratings 0.02;

Result = LIMIT food_ratings_2percent 10;

DUMP Result;
```



```
(Joe,31,16,19,49,4)
(Joe,2,3,3,13,5)
(Joy,42,36,22,39,3)
(Sam,27,10,41,31,1)
(Mel,3,29,36,4,3)
(Jill,36,19,27,32,3)
(Joy,30,35,18,42,1)
(Sam,44,10,46,38,3)
(Sam,37,17,41,19,2)
(Jill,43,33,46,49,1)
grunt>
```

Q 6. Write and execute a sequence of pig latin statements that loads the foodplaces file as a relation. Call the relation 'food_places'. The load command should associate a schema with this relation where the first attribute is referred to as 'placeid' and is of type int and the second attribute is referred to as 'placename' and is of type chararray.

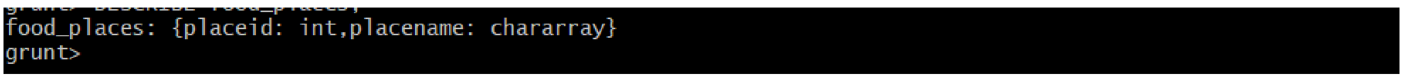
Execute the describe command on this relation.

Now perform a join between the initial place_ratings relation and the food_places relation on the placeid attributes to create a new relation called 'food_ratings_w_place_names'. This new relation should have all the attributes (columns) of both relations. The new relation will allow us to work with place ratings and place names together.

Write 6 records of this relation out to the console.

```
food_places = LOAD '/user/hadoop/foodplaces149222.txt' USING PigStorage(',') AS (placeid:int,
placename: chararray);
```

```
DESCRIBE food_places;
```

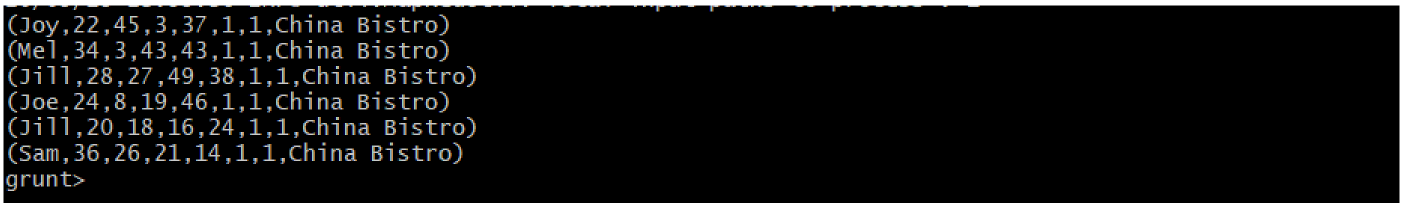


```
food_places: {placeid: int,placename: chararray}
grunt>
```

```
food_ratings_w_place_names = JOIN food_ratings BY placeid, food_places BY placeid;
```

```
Result = LIMIT food_ratings_w_place_names 6;
```

```
DUMP Result;
```



```
(Joy,22,45,3,37,1,1,China Bistro)
(Mel,34,3,43,43,1,1,China Bistro)
(Jill,28,27,49,38,1,1,China Bistro)
(Joe,24,8,19,46,1,1,China Bistro)
(Jill,20,18,16,24,1,1,China Bistro)
(Sam,36,26,21,14,1,1,China Bistro)
grunt>
```

Q 7. Identify the one correct answer for each the following questions. These questions are similar to the ones you might find on the mid-term covering Pig. Each is worth ½ point.

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

Answer: **A**

Choices:

A. LIMIT

B. DISTINCT

C. UNIQUE

D. SAMPLE

II. Which keyword returns only unique rows for a relation when forming a new relation?

Choices:

Answer: **C**

A. SAMPLE

B. FILTER

C. DISTINCT

D. SPLIT

III. 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?

Choices:

Answer: **B**

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)

IV. 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: **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;

V. Pig Latin is a **data flow** language. Select the best choice to fill in the blank.

Choices:

A. functional

B. data flow

C. procedural

D. declarative

VI. 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: **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