CS 595 HOMEWORK 5

E1

Magic Number is 116208

```
 food_ratings = LOAD 'foodratings116208.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}
```

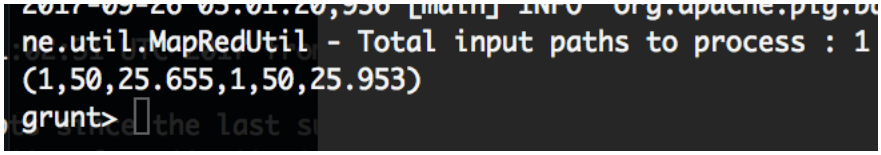
E2

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

STORE food_ratings_subset INTO 'food_ratings_subset.txt' using
PigStorage(',');

A = LIMIT food_ratings_subset 6;

dump A;
```

```
ne.util.MapRedUtil - Total input paths to process : 1
(Mel,31)
(Joe,6)
(Joe,16)
(Mel,31)
(Mel,37)
(Joy,15)
grunt>
```

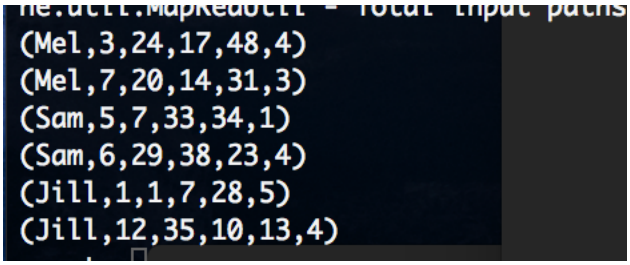
E3

```
food_ratings_profile = GROUP food_ratings ALL;

food_ratings_profile = FOREACH food_ratings_profile 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;
```



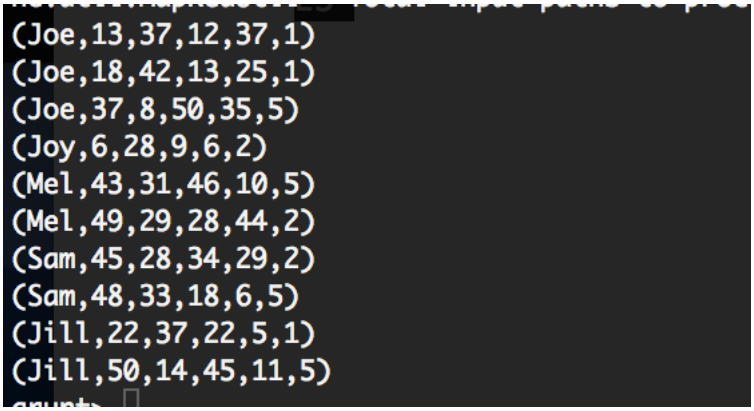
E4

```
food_ratings_filtered = FILTER food_ratings BY f1 < 20 AND f3>5;
food_ratings_filtered = LIMIT food_ratings_filtered 6;
DUMP food_ratings_filtered;
```



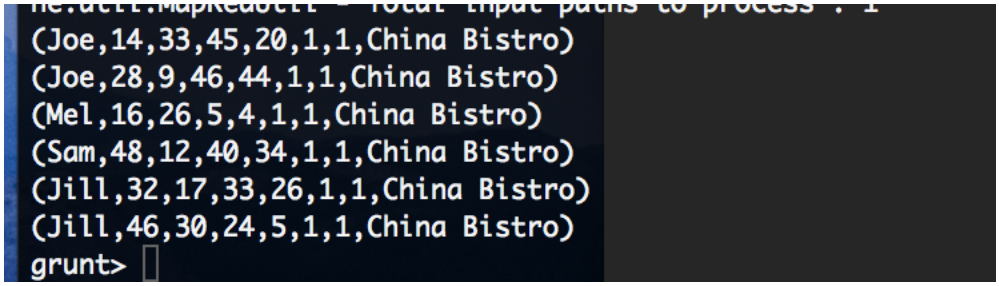
E5

```
grunt> food_ratings_2percent = SAMPLE food_ratings 0.02;
grunt> food_ratings_2percent = LIMIT food_ratings_2percent 10;
grunt> DUMP food_ratings_2percent;
```



E6

```
food_places = LOAD 'foodplaces116208.txt' USING PigStorage(',') AS
(placeid:int, placename:chararray);

food_ratings_w_place_names = JOIN food_ratings BY placeid FULL OUTER,
food_places BY placeid;

 food_ratings_w_place_names = LIMIT food_ratings_w_place_names 6;

 DUMP food_ratings_w_place_names;
```



E7

Resilient Distributed Datasets (RDDs) is a distributed memory abstraction that lets programmers perform in-memory computations on large clusters in a fault-tolerant manner. RDDs are motivated by two types of applications that current computing frameworks handle inefficiently: iterative algorithms and interactive data mining tools. In both cases, keeping data in memory can improve performance by an order of magnitude. To achieve fault tolerance efficiently, RDDs provide a restricted form of shared memory, based on coarse-grained transformations rather than fine-grained updates to shared state. However, RDDs are expressive enough to capture a wide class of computations, including recent specialized programming models for iterative jobs, such as Pregel, and new applications that these models do not capture. Authors have implemented RDDs in a system called Spark, which they evaluate through a variety of user applications and benchmarks.

MapReduce is an exiting solution but it has disadvantages: huge memory consumption, batch processing orientation(interactive problem) as well as Pregel, which only support for specific computation patterns. Spark is an efficient way to generally share data across multiple computations which stands for a fast and general engine for large-scale data processing.