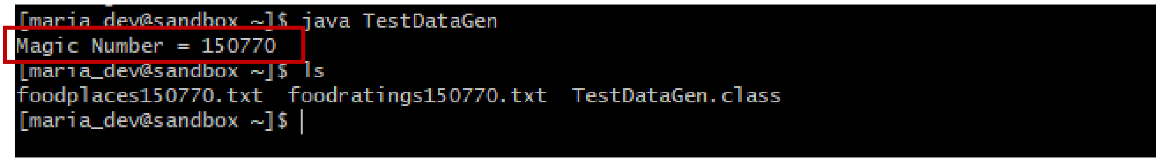
CSP595 - Assignment 5

Name: Jagruti Vichare
Email ID: jvichare@hawk.iit.edu
CWID: A20378092

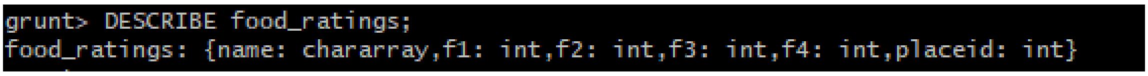
Exercise 1)



```
[maria_dev@sandbox ~]$ java TestDataGen
Magic Number = 150770
[maria_dev@sandbox ~]$ ls
foodplaces150770.txt  foodratings150770.txt  TestDataGen.class
[maria_dev@sandbox ~]$ |
```

Commands:

1) food_ratings = LOAD '/user/maria_dev/ foodratings150770.txt' USING PigStorage(',') AS (name:chararray, f1:int, f2:int, f3:int, f4:int, placeid:int);
2) DESCRIBE food_ratings;

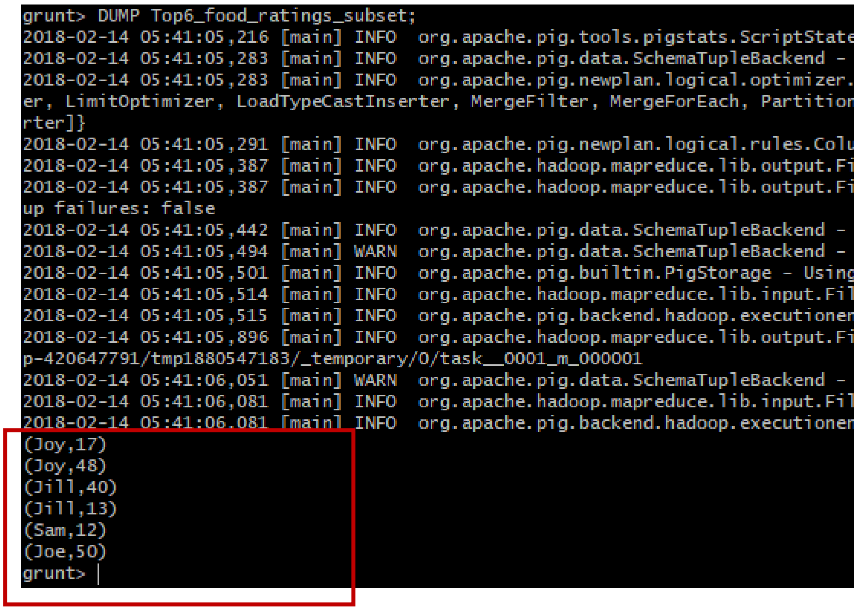


```
grunt> DESCRIBE food_ratings;
food_ratings: {name: chararray,f1: int,f2: int,f3: int,f4: int,placeid: int}
```

Exercise 2)

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;

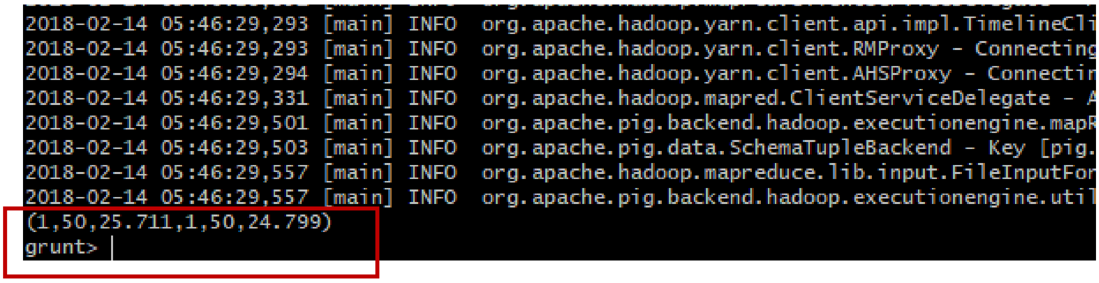


```
grunt> DUMP Top6_food_ratings_subset;
2018-02-14 05:41:05,216 [main] INFO  org.apache.pig.tools.pigstats.ScriptState
2018-02-14 05:41:05,283 [main] INFO  org.apache.pig.data.SchemaTupleBackend -
2018-02-14 05:41:05,283 [main] INFO  org.apache.pig.newplan.logical.optimizer.
er, LimitOptimizer, LoadTypeCastInserter, MergeFilter, MergeForEach, Partition
rter]}
2018-02-14 05:41:05,291 [main] INFO  org.apache.pig.newplan.logical.rules.Colu
2018-02-14 05:41:05,387 [main] INFO  org.apache.hadoop.mapreduce.lib.output.Fi
2018-02-14 05:41:05,387 [main] INFO  org.apache.hadoop.mapreduce.lib.output.Fi
up failures: false
2018-02-14 05:41:05,442 [main] INFO  org.apache.pig.data.SchemaTupleBackend -
2018-02-14 05:41:05,494 [main] WARN  org.apache.pig.data.SchemaTupleBackend -
2018-02-14 05:41:05,501 [main] INFO  org.apache.pig.builtin.PigStorage - Using
2018-02-14 05:41:05,514 [main] INFO  org.apache.hadoop.mapreduce.lib.input.Fil
2018-02-14 05:41:05,515 [main] INFO  org.apache.pig.backend.hadoop.executioner
2018-02-14 05:41:05,896 [main] INFO  org.apache.hadoop.mapreduce.lib.output.Fi
p-420647791/tmp1880547183/_temporary/0/task__0001_m_000001
2018-02-14 05:41:06,051 [main] WARN  org.apache.pig.data.SchemaTupleBackend -
2018-02-14 05:41:06,081 [main] INFO  org.apache.hadoop.mapreduce.lib.input.Fil
2018-02-14 05:41:06,081 [main] INFO  org.apache.pig.backend.hadoop.executioner
(Joy,17)
(Joy,48)
(Jill,40)
(Jill,13)
(Sam,12)
(Joe,50)
grunt> |
```

Exercise 3)

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;

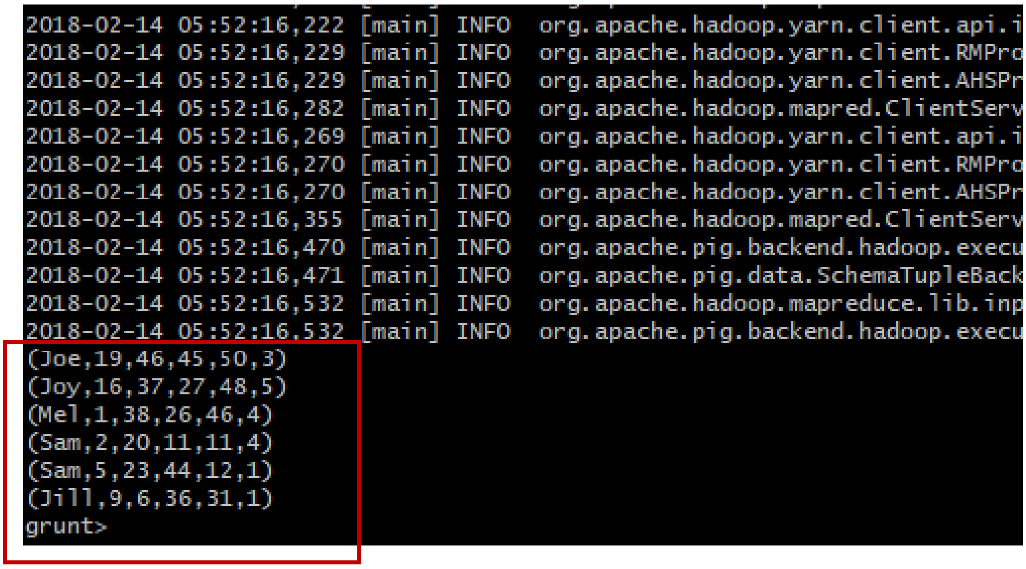


```
2018-02-14 05:46:29,293 [main] INFO  org.apache.hadoop.yarn.client.api.impl.TimelineCli
2018-02-14 05:46:29,293 [main] INFO  org.apache.hadoop.yarn.client.RMProxy - Connecting
2018-02-14 05:46:29,294 [main] INFO  org.apache.hadoop.yarn.client.AHSProxy - Connectin
2018-02-14 05:46:29,331 [main] INFO  org.apache.hadoop.mapred.ClientServiceDelegate - A
2018-02-14 05:46:29,501 [main] INFO  org.apache.pig.backend.hadoop.executionengine.mapR
2018-02-14 05:46:29,503 [main] INFO  org.apache.pig.data.SchemaTupleBackend - Key [pig.
2018-02-14 05:46:29,557 [main] INFO  org.apache.hadoop.mapreduce.lib.input.FileInputFor
2018-02-14 05:46:29,557 [main] INFO  org.apache.pig.backend.hadoop.executionengine.util
(1,50,25.711,1,50,24.799)
grunt> 
```

Exercise 4)

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;

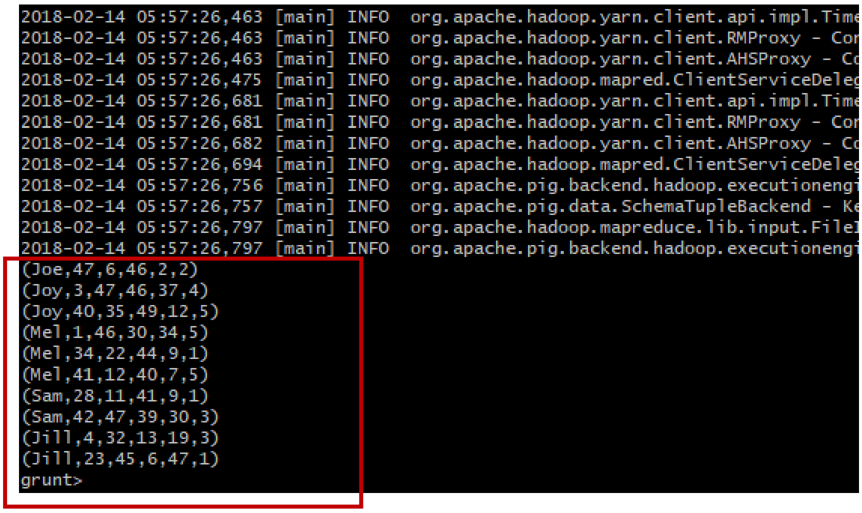


```
2018-02-14 05:52:16,222 [main] INFO  org.apache.hadoop.yarn.client.api.i
2018-02-14 05:52:16,229 [main] INFO  org.apache.hadoop.yarn.client.RMPro
2018-02-14 05:52:16,229 [main] INFO  org.apache.hadoop.yarn.client.AHSPr
2018-02-14 05:52:16,282 [main] INFO  org.apache.hadoop.mapred.ClientServ
2018-02-14 05:52:16,269 [main] INFO  org.apache.hadoop.yarn.client.api.i
2018-02-14 05:52:16,270 [main] INFO  org.apache.hadoop.yarn.client.RMPro
2018-02-14 05:52:16,270 [main] INFO  org.apache.hadoop.yarn.client.AHSPr
2018-02-14 05:52:16,355 [main] INFO  org.apache.hadoop.mapred.ClientServ
2018-02-14 05:52:16,470 [main] INFO  org.apache.pig.backend.hadoop.execu
2018-02-14 05:52:16,471 [main] INFO  org.apache.pig.data.SchemaTupleBack
2018-02-14 05:52:16,532 [main] INFO  org.apache.hadoop.mapreduce.lib.inp
2018-02-14 05:52:16,532 [main] INFO  org.apache.pig.backend.hadoop.execu
(Joe,19,46,45,50,3)
(Joy,16,37,27,48,5)
(Mel,1,38,26,46,4)
(Sam,2,20,11,11,4)
(Sam,5,23,44,12,1)
(Jill,9,6,36,31,1)
grunt>
```

Exercise 5)

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;

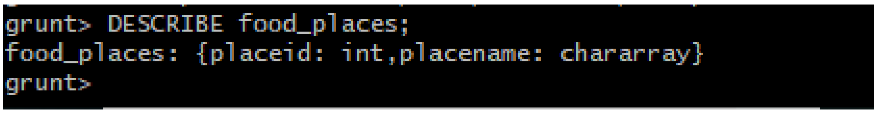


```
2018-02-14 05:57:26,463 [main] INFO  org.apache.hadoop.yarn.client.api.impl.Time
2018-02-14 05:57:26,463 [main] INFO  org.apache.hadoop.yarn.client.RMProxy - Cor
2018-02-14 05:57:26,463 [main] INFO  org.apache.hadoop.yarn.client.AHSProxy - Co
2018-02-14 05:57:26,475 [main] INFO  org.apache.hadoop.mapred.ClientServiceDeleg
2018-02-14 05:57:26,681 [main] INFO  org.apache.hadoop.yarn.client.api.impl.Time
2018-02-14 05:57:26,681 [main] INFO  org.apache.hadoop.yarn.client.RMProxy - Cor
2018-02-14 05:57:26,682 [main] INFO  org.apache.hadoop.yarn.client.AHSProxy - Co
2018-02-14 05:57:26,694 [main] INFO  org.apache.hadoop.mapred.ClientServiceDeleg
2018-02-14 05:57:26,756 [main] INFO  org.apache.pig.backend.hadoop.executioneng
2018-02-14 05:57:26,757 [main] INFO  org.apache.pig.data.SchemaTupleBackend - Ke
2018-02-14 05:57:26,797 [main] INFO  org.apache.hadoop.mapreduce.lib.input.FileI
2018-02-14 05:57:26,797 [main] INFO  org.apache.pig.backend.hadoop.executioneng
(Joe,47,6,46,2,2)
(Joy,3,47,46,37,4)
(Joy,40,35,49,12,5)
(Mel,1,46,30,34,5)
(Mel,34,22,44,9,1)
(Mel,41,12,40,7,5)
(Sam,28,11,41,9,1)
(Sam,42,47,39,30,3)
(Jill,4,32,13,19,3)
(Jill,23,45,6,47,1)
grunt>
```

Exercise 6)

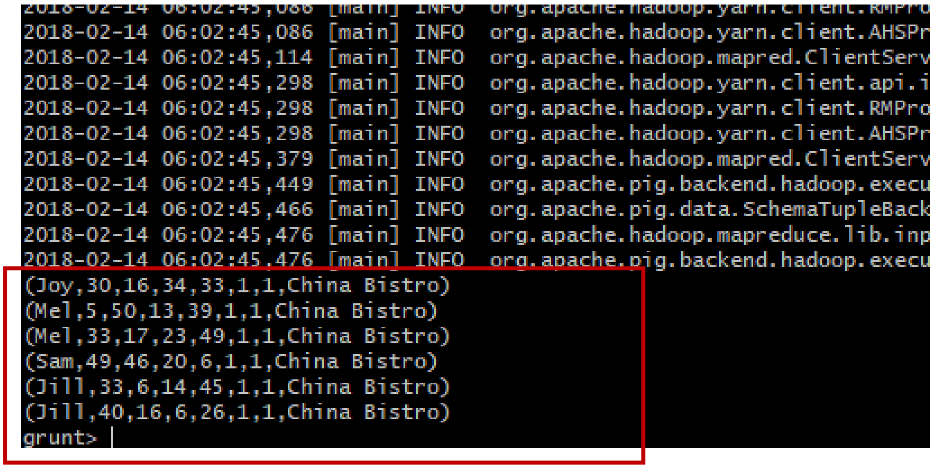
Commands:

1) food_places = LOAD '/user/maria_dev/foodplaces150770.txt' USING PigStorage(',') AS (placeid:int, placename:chararray);
2) DESCRIBE food_places;



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

3) food_ratings_w_place_names = JOIN food_ratings BY placeid, food_places BY placeid;
4) Top6_food_ratings_w_place_names = LIMIT food_ratings_w_place_names 6;
5) DUMP Top6_food_ratings_w_place_names;



```
2018-02-14 06:02:45,086 [main] INFO  org.apache.hadoop.yarn.client.RMPro
2018-02-14 06:02:45,086 [main] INFO  org.apache.hadoop.yarn.client.AHSPr
2018-02-14 06:02:45,114 [main] INFO  org.apache.hadoop.mapred.ClientServ
2018-02-14 06:02:45,298 [main] INFO  org.apache.hadoop.yarn.client.api.i
2018-02-14 06:02:45,298 [main] INFO  org.apache.hadoop.yarn.client.RMPro
2018-02-14 06:02:45,298 [main] INFO  org.apache.hadoop.yarn.client.AHSPr
2018-02-14 06:02:45,379 [main] INFO  org.apache.hadoop.mapred.ClientServ
2018-02-14 06:02:45,449 [main] INFO  org.apache.pig.backend.hadoop.execu
2018-02-14 06:02:45,466 [main] INFO  org.apache.pig.data.SchemaTupleBack
2018-02-14 06:02:45,476 [main] INFO  org.apache.hadoop.mapreduce.lib.inp
2018-02-14 06:02:45,476 [main] INFO  org.apache.pig.backend.hadoop.execu
(Joy,30,16,34,33,1,1,China Bistro)
(Mel,5,50,13,39,1,1,China Bistro)
(Mel,33,17,23,49,1,1,China Bistro)
(Sam,49,46,20,6,1,1,China Bistro)
(Jill,33,6,14,45,1,1,China Bistro)
(Jill,40,16,6,26,1,1,China Bistro)
grunt>
```

Exercise 7) (Extra credit)

The paper describes Resilient Distributed Datasets (RDDs), how they can be used for in-memory computation, how they can solve problems which existing cluster computing frameworks like MapReduce and Dryad can not resolve.

Current cluster computing frameworks are inefficient when it comes to:

1. Iterative algorithms – Reuse intermediate results across multiple computations
2. Interactive data mining tools – Run multiple ad-hoc queries on the same subset of the data

In both cases, keeping data in memory can improve performance by an order of magnitude. That is when RDDs came into picture. RDDs are fault-tolerant, parallel data structures that let users explicitly persist intermediate results in memory, control their partitioning to optimize data placement, and manipulate them using a rich set of operators. RDDs can express a wide range of parallel applications, including many specialized programming models that have been proposed for iterative computation, and new applications that these models do not capture. Unlike existing storage abstractions for clusters, which require data replication for fault tolerance, RDDs offer an API based on coarse-grained transformations that lets them recover data efficiently using lineage. RDDs have been implemented in Spark which outperforms Hadoop by up to 20x in iterative applications and can be used interactively to query hundreds of gigabytes of data.

Comments: RDDs can be used in place of existing models like MapReduce, Pregel, Dryad etc. RDDs offer limited interface due to their coarse-grained transformations but these limitations have very small impact on many parallel applications. The previous frameworks have not offered the same level of generality because they lacked data sharing abstractions.