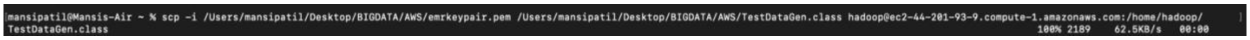
CSP554—Big Data Technologies

Assignment #5 (Modules 05)

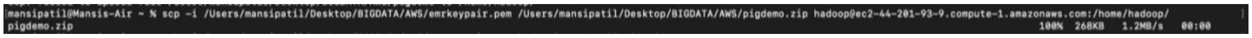
Worth: 15 points

Exercise 1)

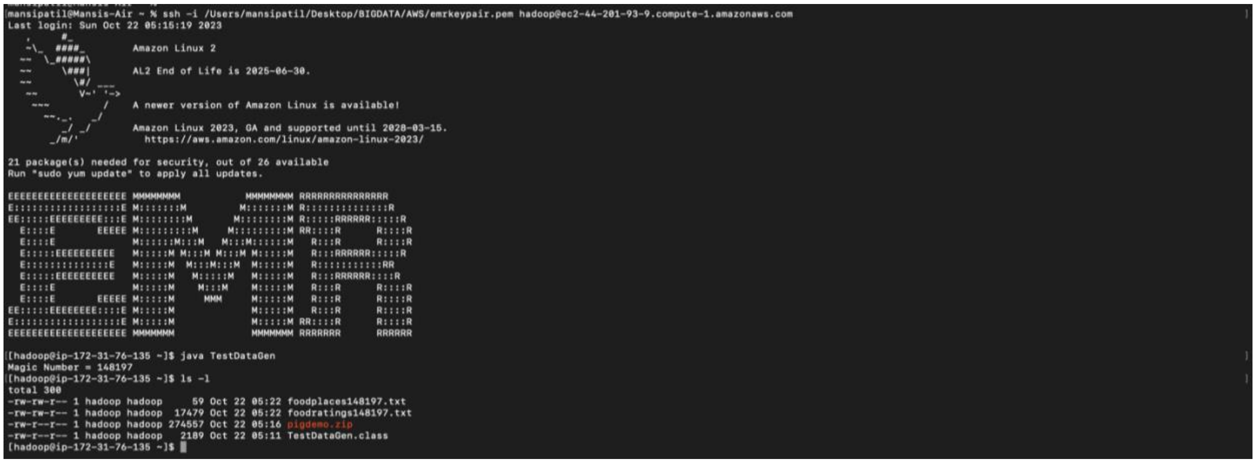
Magic Number Generation –



```
mansipatil@Mansis-Air ~ % scp -i /Users/mansipatil/Desktop/BIGDATA/AWS/emrkeypair.pem /Users/mansipatil/Desktop/BIGDATA/AWS/TestDataGen.class hadoop@ec2-44-201-93-9.compute-1.amazonaws.com:/home/hadoop/
TestDataGen.class                                                                                                             100% 2189     62.5KB/s   00:00
```



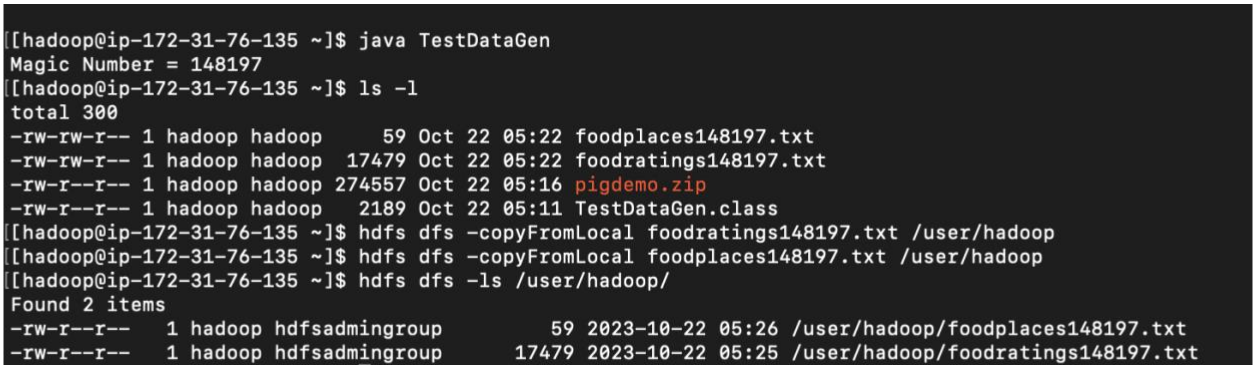
```
mansipatil@Mansis-Air ~ % scp -i /Users/mansipatil/Desktop/BIGDATA/AWS/emrkeypair.pem /Users/mansipatil/Desktop/BIGDATA/AWS/pigdemo.zip hadoop@ec2-44-201-93-9.compute-1.amazonaws.com:/home/hadoop/
pigdemo.zip                                                                                                                   100%  268KB   1.2MB/s   00:00
```



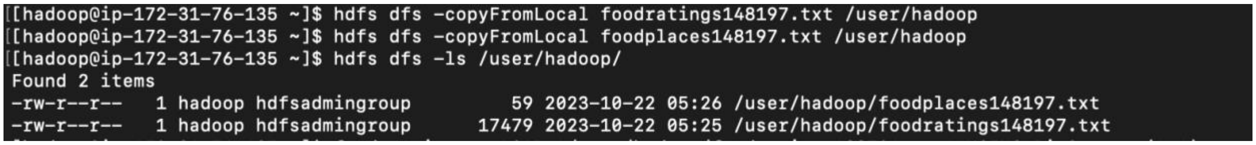
```
mansipatil@Mansis-Air ~ % ssh -i /Users/mansipatil/Desktop/BIGDATA/AWS/emrkeypair.pem hadoop@ec2-44-201-93-9.compute-1.amazonaws.com
Last login: Sun Oct 22 05:15:19 2023
   ,     #_
   ~\_  ####_        Amazon Linux 2
  ~~  \_#####\
  ~~     \###|       AL2 End of Life is 2025-06-30.
  ~~       \#/ ___
   ~~       V~' '->
    ~~~         /    A newer version of Amazon Linux is available!
      ~~._.   _/
         _/ _/       Amazon Linux 2023, GA and supported until 2028-03-15.
       _/m/'           https://aws.amazon.com/linux/amazon-linux-2023/

21 package(s) needed for security, out of 26 available
Run "sudo yum update" to apply all updates.

EEEEEEEEEEEEEEEEEEEE MMMMMMMM           MMMMMMMM RRRRRRRRRRRRRRR
E::::::::::::::::::E M:::::::M         M:::::::M R::::::::::::::R
EE:::::EEEEEEEEE:::E M::::::::M       M::::::::M R:::::RRRRRR:::::R
  E::::E       EEEEE M:::::::::M     M:::::::::M RR::::R      R::::R
  E::::E             M::::::M:::M   M:::M::::::M   R:::R      R::::R
  E:::::EEEEEEEEEE   M:::::M M:::M M:::M M:::::M   R:::RRRRRR:::::R
  E::::::::::::::E   M:::::M  M:::M:::M  M:::::M   R:::::::::::RR
  E:::::EEEEEEEEEE   M:::::M   M:::::M   M:::::M   R:::RRRRRR::::R
  E::::E             M:::::M    M:::M    M:::::M   R:::R      R::::R
  E::::E       EEEEE M:::::M     MMM     M:::::M   R:::R      R::::R
EE:::::EEEEEEEE::::E M:::::M             M:::::M   R:::R      R::::R
E::::::::::::::::::E M:::::M             M:::::M RR::::R      R::::R
EEEEEEEEEEEEEEEEEEEE MMMMMMM             MMMMMMM RRRRRRR      RRRRRR

[hadoop@ip-172-31-76-135 ~]$ java TestDataGen
Magic Number = 148197
[hadoop@ip-172-31-76-135 ~]$ ls -l
total 300
-rw-rw-r-- 1 hadoop hadoop     59 Oct 22 05:22 foodplaces148197.txt
-rw-rw-r-- 1 hadoop hadoop  17479 Oct 22 05:22 foodratings148197.txt
-rw-r--r-- 1 hadoop hadoop 274557 Oct 22 05:16 pigdemo.zip
-rw-r--r-- 1 hadoop hadoop   2189 Oct 22 05:11 TestDataGen.class
[hadoop@ip-172-31-76-135 ~]$ 
```

Magic Number – **148197**

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



```
[hadoop@ip-172-31-76-135 ~]$ java TestDataGen
Magic Number = 148197
[hadoop@ip-172-31-76-135 ~]$ ls -l
total 300
-rw-rw-r-- 1 hadoop hadoop     59 Oct 22 05:22 foodplaces148197.txt
-rw-rw-r-- 1 hadoop hadoop  17479 Oct 22 05:22 foodratings148197.txt
-rw-r--r-- 1 hadoop hadoop 274557 Oct 22 05:16 pigdemo.zip
-rw-r--r-- 1 hadoop hadoop   2189 Oct 22 05:11 TestDataGen.class
[hadoop@ip-172-31-76-135 ~]$ hdfs dfs -copyFromLocal foodratings148197.txt /user/hadoop
[hadoop@ip-172-31-76-135 ~]$ hdfs dfs -copyFromLocal foodplaces148197.txt /user/hadoop
[hadoop@ip-172-31-76-135 ~]$ hdfs dfs -ls /user/hadoop/
Found 2 items
-rw-r--r--   1 hadoop hdfsadmingroup         59 2023-10-22 05:26 /user/hadoop/foodplaces148197.txt
-rw-r--r--   1 hadoop hdfsadmingroup      17479 2023-10-22 05:25 /user/hadoop/foodratings148197.txt
```



```
[hadoop@ip-172-31-76-135 ~]$ hdfs dfs -copyFromLocal foodratings148197.txt /user/hadoop
[hadoop@ip-172-31-76-135 ~]$ hdfs dfs -copyFromLocal foodplaces148197.txt /user/hadoop
[hadoop@ip-172-31-76-135 ~]$ hdfs dfs -ls /user/hadoop/
Found 2 items
-rw-r--r--   1 hadoop hdfsadmingroup         59 2023-10-22 05:26 /user/hadoop/foodplaces148197.txt
-rw-r--r--   1 hadoop hdfsadmingroup      17479 2023-10-22 05:25 /user/hadoop/foodratings148197.txt
```

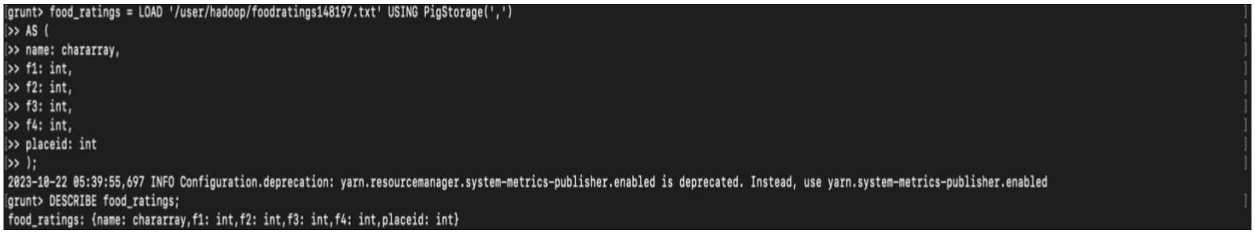
Write and execute a sequence of pig latin statements that loads the foodratings file as a relation. Call the relation ‘food_ratings’. The load command should associate a schema with this relation where the first attribute is referred to as ‘name’ and is of type chararray, the next attributes are referred 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.

ANS:

MAGIC NUMBER: 148197

COMMAND:

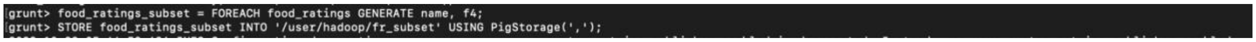
```
food_ratings = LOAD '/user/hadoop/foodratings148197.txt' USING PigStorage(',')
AS (
name: chararray,
f1: int,
f2: int,
f3: int,
f4: int,
placeid: int
);
DESCRIBE food_ratings;
```



```
grunt> food_ratings = LOAD '/user/hadoop/foodratings148197.txt' USING PigStorage(',')
>> AS (
>> name: chararray,
>> f1: int,
>> f2: int,
>> f3: int,
>> f4: int,
>> placeid: int
>> );
2023-10-22 05:39:55,697 INFO Configuration.deprecation: yarn.resourcemanager.system-metrics-publisher.enabled is deprecated. Instead, use yarn.system-metrics-publisher.enabled
grunt> DESCRIBE food_ratings;
food_ratings: {name: chararray,f1: int,f2: int,f3: int,f4: int,placeid: int}
```

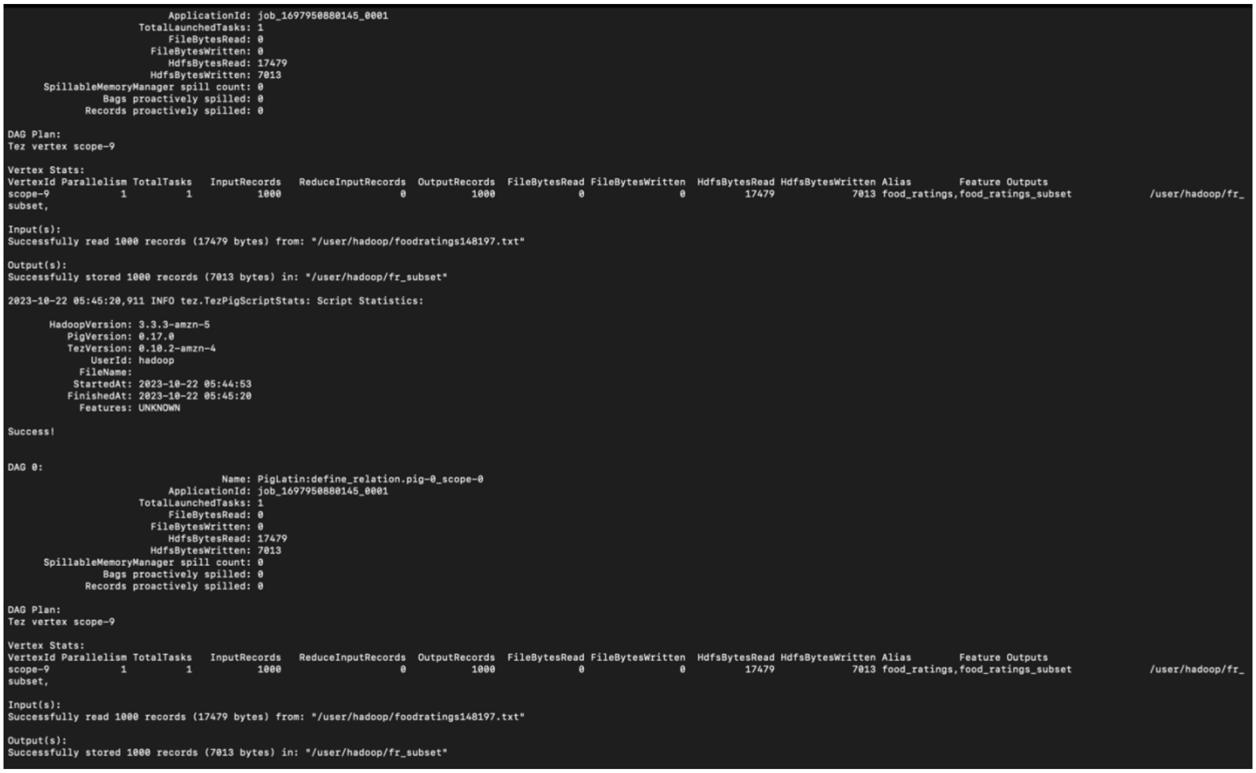
Exercise 2)

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



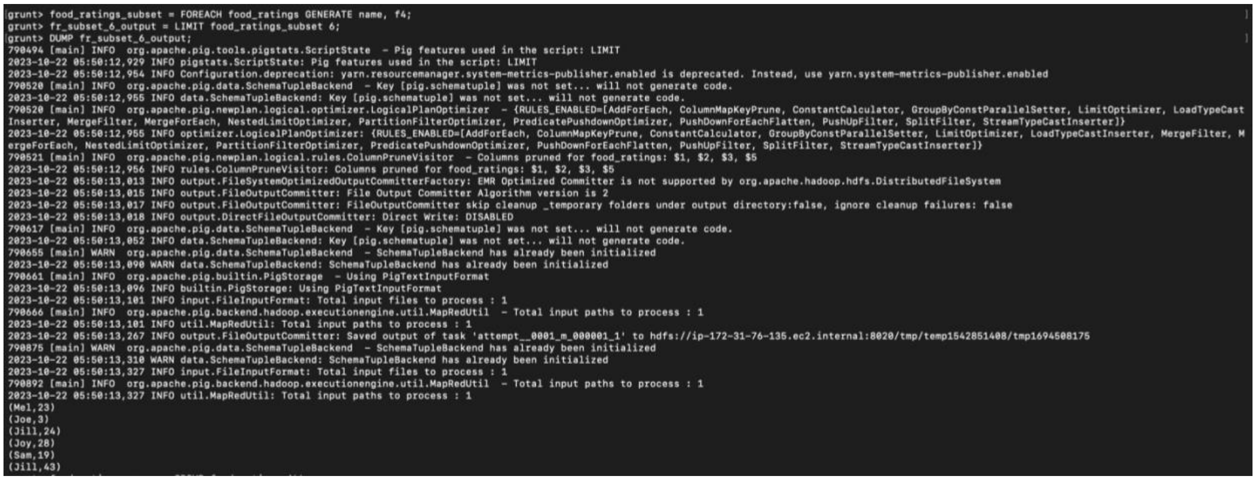
```
grunt> food_ratings_subset = FOREACH food_ratings GENERATE name, f4;
grunt> STORE food_ratings_subset INTO '/user/hadoop/fr_subset' USING PigStorage(',');
```

Store this last relation, food_ratings_subset, back to HDFS (perhaps as the file /user/hadoop/fr_subset)



```
                     ApplicationId: job_1697950880145_0001
                 TotalLaunchedTasks: 1
                      FileBytesRead: 0
                   FileBytesWritten: 0
                      HdfsBytesRead: 17479
                   HdfsBytesWritten: 7013
      SpillableMemoryManager spill count: 0
                Bags proactively spilled: 0
             Records proactively spilled: 0

DAG Plan:
Tez vertex scope-9

Vertex Stats:
VertexId Parallelism TotalTasks   InputRecords   ReduceInputRecords  OutputRecords  FileBytesRead FileBytesWritten  HdfsBytesRead HdfsBytesWritten Alias        Feature Outputs
scope-9            1          1           1000                    0           1000              0                0          17479             7013 food_ratings,food_ratings_subset            /user/hadoop/fr_
subset,

Input(s):
Successfully read 1000 records (17479 bytes) from: "/user/hadoop/foodratings148197.txt"

Output(s):
Successfully stored 1000 records (7013 bytes) in: "/user/hadoop/fr_subset"

2023-10-22 05:45:20,911 INFO tez.TezPigScriptStats: Script Statistics:

       HadoopVersion: 3.3.3-amzn-5
          PigVersion: 0.17.0
          TezVersion: 0.10.2-amzn-4
              UserId: hadoop
            FileName: 
           StartedAt: 2023-10-22 05:44:53
          FinishedAt: 2023-10-22 05:45:20
            Features: UNKNOWN

Success!


DAG 0:
                              Name: PigLatin:define_relation.pig-0_scope-0
                     ApplicationId: job_1697950880145_0001
                 TotalLaunchedTasks: 1
                      FileBytesRead: 0
                   FileBytesWritten: 0
                      HdfsBytesRead: 17479
                   HdfsBytesWritten: 7013
      SpillableMemoryManager spill count: 0
                Bags proactively spilled: 0
             Records proactively spilled: 0

DAG Plan:
Tez vertex scope-9

Vertex Stats:
VertexId Parallelism TotalTasks   InputRecords   ReduceInputRecords  OutputRecords  FileBytesRead FileBytesWritten  HdfsBytesRead HdfsBytesWritten Alias        Feature Outputs
scope-9            1          1           1000                    0           1000              0                0          17479             7013 food_ratings,food_ratings_subset            /user/hadoop/fr_
subset,

Input(s):
Successfully read 1000 records (17479 bytes) from: "/user/hadoop/foodratings148197.txt"

Output(s):
Successfully stored 1000 records (7013 bytes) in: "/user/hadoop/fr_subset"
```

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



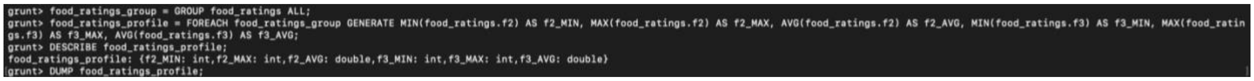
```
grunt> food_ratings_subset = FOREACH food_ratings GENERATE name, f4;
grunt> fr_subset_6_output = LIMIT food_ratings_subset 6;
grunt> DUMP fr_subset_6_output;
790494 [main] INFO  org.apache.pig.tools.pigstats.ScriptState  - Pig features used in the script: LIMIT
2023-10-22 05:50:12,929 INFO pigstats.ScriptState: Pig features used in the script: LIMIT
2023-10-22 05:50:12,954 INFO Configuration.deprecation: yarn.resourcemanager.system-metrics-publisher.enabled is deprecated. Instead, use yarn.system-metrics-publisher.enabled
790520 [main] INFO  org.apache.pig.data.SchemaTupleBackend  - Key [pig.schematuple] was not set... will not generate code.
2023-10-22 05:50:12,955 INFO data.SchemaTupleBackend: Key [pig.schematuple] was not set... will not generate code.
790520 [main] INFO  org.apache.pig.newplan.logical.optimizer.LogicalPlanOptimizer  - {RULES_ENABLED=[AddForEach, ColumnMapKeyPrune, ConstantCalculator, GroupByConstParallelSetter, LimitOptimizer, LoadTypeCast
Inserter, MergeFilter, MergeForEach, NestedLimitOptimizer, PartitionFilterOptimizer, PredicatePushdownOptimizer, PushDownForEachFlatten, PushUpFilter, SplitFilter, StreamTypeCastInserter]}
2023-10-22 05:50:12,955 INFO optimizer.LogicalPlanOptimizer: {RULES_ENABLED=[AddForEach, ColumnMapKeyPrune, ConstantCalculator, GroupByConstParallelSetter, LimitOptimizer, LoadTypeCastInserter, MergeFilter, M
ergeForEach, NestedLimitOptimizer, PartitionFilterOptimizer, PredicatePushdownOptimizer, PushDownForEachFlatten, PushUpFilter, SplitFilter, StreamTypeCastInserter]}
790521 [main] INFO  org.apache.pig.newplan.logical.rules.ColumnPruneVisitor  - Columns pruned for food_ratings: $1, $2, $3, $5
2023-10-22 05:50:12,956 INFO rules.ColumnPruneVisitor: Columns pruned for food_ratings: $1, $2, $3, $5
2023-10-22 05:50:13,013 INFO output.FileSystemOptimizedOutputCommitterFactory: EMR Optimized Committer is not supported by org.apache.hadoop.hdfs.DistributedFileSystem
2023-10-22 05:50:13,015 INFO output.FileOutputCommitter: File Output Committer Algorithm version is 2
2023-10-22 05:50:13,017 INFO output.FileOutputCommitter: FileOutputCommitter skip cleanup _temporary folders under output directory:false, ignore cleanup failures: false
2023-10-22 05:50:13,018 INFO output.DirectFileOutputCommitter: Direct Write: DISABLED
790617 [main] INFO  org.apache.pig.data.SchemaTupleBackend  - Key [pig.schematuple] was not set... will not generate code.
2023-10-22 05:50:13,052 INFO data.SchemaTupleBackend: Key [pig.schematuple] was not set... will not generate code.
790655 [main] WARN  org.apache.pig.data.SchemaTupleBackend  - SchemaTupleBackend has already been initialized
2023-10-22 05:50:13,090 WARN data.SchemaTupleBackend: SchemaTupleBackend has already been initialized
790661 [main] INFO  org.apache.pig.builtin.PigStorage  - Using PigTextInputFormat
2023-10-22 05:50:13,096 INFO builtin.PigStorage: Using PigTextInputFormat
2023-10-22 05:50:13,101 INFO input.FileInputFormat: Total input files to process : 1
790666 [main] INFO  org.apache.pig.backend.hadoop.executionengine.util.MapRedUtil  - Total input paths to process : 1
2023-10-22 05:50:13,101 INFO util.MapRedUtil: Total input paths to process : 1
2023-10-22 05:50:13,267 INFO output.FileOutputCommitter: Saved output of task 'attempt__0001_m_000001_1' to hdfs://ip-172-31-76-135.ec2.internal:8020/tmp/temp1542851408/tmp1694508175
790875 [main] WARN  org.apache.pig.data.SchemaTupleBackend  - SchemaTupleBackend has already been initialized
2023-10-22 05:50:13,310 WARN data.SchemaTupleBackend: SchemaTupleBackend has already been initialized
2023-10-22 05:50:13,327 INFO input.FileInputFormat: Total input files to process : 1
790892 [main] INFO  org.apache.pig.backend.hadoop.executionengine.util.MapRedUtil  - Total input paths to process : 1
2023-10-22 05:50:13,327 INFO util.MapRedUtil: Total input paths to process : 1
(Mel,23)
(Joe,3)
(Jill,24)
(Joy,28)
(Sam,19)
(Jill,43)
```

Commands –

```
food_ratings_subset = FOREACH food_ratings GENERATE name, f4;
STORE food_ratings_subset INTO '/user/hadoop/fr_subset' USING PigStorage(',');
fr_subset_6_output = LIMIT food_ratings_subset 6;
DUMP fr_subset_6_output;
```

Exercise 3)

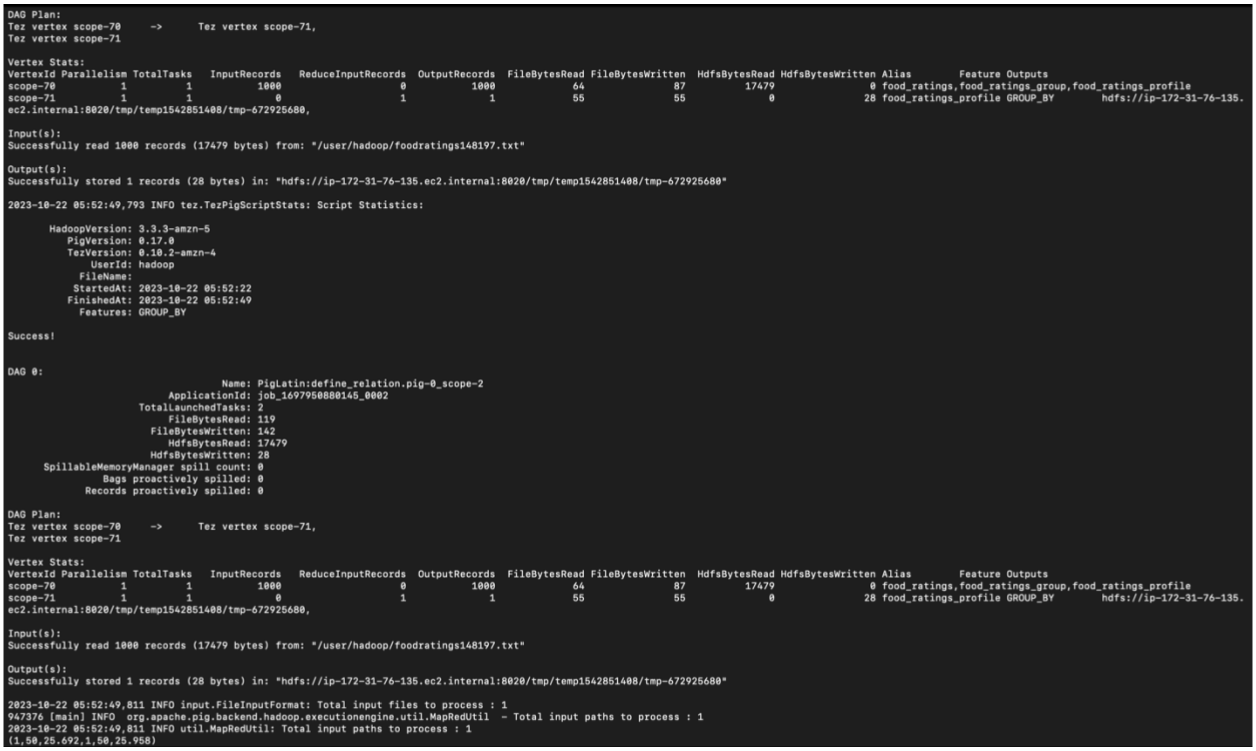
Now create another relation using the initial (food_ratings) relation. Call this relation 'food_ratings_profile'. The new relation should only have one record. This record should hold the minimum, maximum and average values for the attributes 'f2' and 'f3'. (So this one record will have 6 fileds).



```
grunt> food_ratings_group = GROUP food_ratings ALL;
grunt> food_ratings_profile = FOREACH food_ratings_group GENERATE MIN(food_ratings.f2) AS f2_MIN, MAX(food_ratings.f2) AS f2_MAX, AVG(food_ratings.f2) AS f2_AVG, MIN(food_ratings.f3) AS f3_MIN, MAX(food_ratin
gs.f3) AS f3_MAX, AVG(food_ratings.f3) AS f3_AVG;
grunt> DESCRIBE food_ratings_profile;
food_ratings_profile: {f2_MIN: int,f2_MAX: int,f2_AVG: double,f3_MIN: int,f3_MAX: int,f3_AVG: double}
grunt> DUMP food_ratings_profile;
```

Commands –

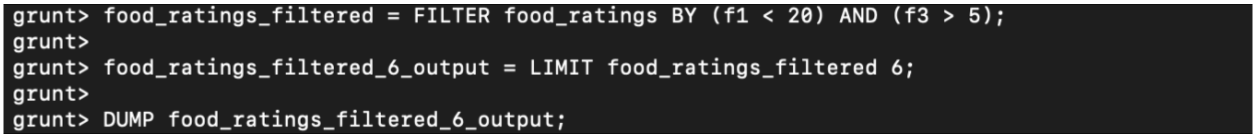
```
food_ratings_group = GROUP food_ratings ALL;
food_ratings_profile = FOREACH food_ratings_group GENERATE MIN(food_ratings.f2) AS f2_MIN, MAX(food_ratings.f2) AS f2_MAX, AVG(food_ratings.f2) AS f2_AVG, MIN(food_ratings.f3) AS f3_MIN, MAX(food_ratings.f3) AS f3_MAX, AVG(food_ratings.f3) AS f3_AVG;
DESCRIBE food_ratings_profile;
DUMP food_ratings_profile;
```



```
DAG Plan:
Tez vertex scope-70     ->      Tez vertex scope-71,
Tez vertex scope-71

Vertex Stats:
VertexId Parallelism TotalTasks   InputRecords   ReduceInputRecords  OutputRecords  FileBytesRead FileBytesWritten  HdfsBytesRead HdfsBytesWritten Alias       Feature Outputs
scope-70           1          1           1000                    0           1000            64               87           17479                0 food_ratings,food_ratings_group,food_ratings_profile
scope-71           1          1              0                    1              1            55               55               0               28 food_ratings_profile GROUP_BY        hdfs://ip-172-31-76-135.
ec2.internal:8020/tmp/temp1542851408/tmp-672925680,

Input(s):
Successfully read 1000 records (17479 bytes) from: "/user/hadoop/foodratings148197.txt"

Output(s):
Successfully stored 1 records (28 bytes) in: "hdfs://ip-172-31-76-135.ec2.internal:8020/tmp/temp1542851408/tmp-672925680"

2023-10-22 05:52:49,793 INFO tez.TezPigScriptStats: Script Statistics:

       HadoopVersion: 3.3.3-amzn-5
          PigVersion: 0.17.0
          TezVersion: 0.10.2-amzn-4
              UserId: hadoop
            FileName:
           StartedAt: 2023-10-22 05:52:22
          FinishedAt: 2023-10-22 05:52:49
            Features: GROUP_BY

Success!


DAG 0:
                                    Name: PigLatin:define_relation.pig-0_scope-2
                           ApplicationId: job_1697950880145_0002
                      TotalLaunchedTasks: 2
                           FileBytesRead: 119
                        FileBytesWritten: 142
                           HdfsBytesRead: 17479
                        HdfsBytesWritten: 28
      SpillableMemoryManager spill count: 0
                Bags proactively spilled: 0
             Records proactively spilled: 0

DAG Plan:
Tez vertex scope-70     ->      Tez vertex scope-71,
Tez vertex scope-71

Vertex Stats:
VertexId Parallelism TotalTasks   InputRecords   ReduceInputRecords  OutputRecords  FileBytesRead FileBytesWritten  HdfsBytesRead HdfsBytesWritten Alias       Feature Outputs
scope-70           1          1           1000                    0           1000            64               87           17479                0 food_ratings,food_ratings_group,food_ratings_profile
scope-71           1          1              0                    1              1            55               55               0               28 food_ratings_profile GROUP_BY        hdfs://ip-172-31-76-135.
ec2.internal:8020/tmp/temp1542851408/tmp-672925680,

Input(s):
Successfully read 1000 records (17479 bytes) from: "/user/hadoop/foodratings148197.txt"

Output(s):
Successfully stored 1 records (28 bytes) in: "hdfs://ip-172-31-76-135.ec2.internal:8020/tmp/temp1542851408/tmp-672925680"

2023-10-22 05:52:49,811 INFO input.FileInputFormat: Total input files to process : 1
947376 [main] INFO  org.apache.pig.backend.hadoop.executionengine.util.MapRedUtil  - Total input paths to process : 1
2023-10-22 05:52:49,811 INFO util.MapRedUtil: Total input paths to process : 1
(1,50,25.692,1,50,25.958)
```

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'.

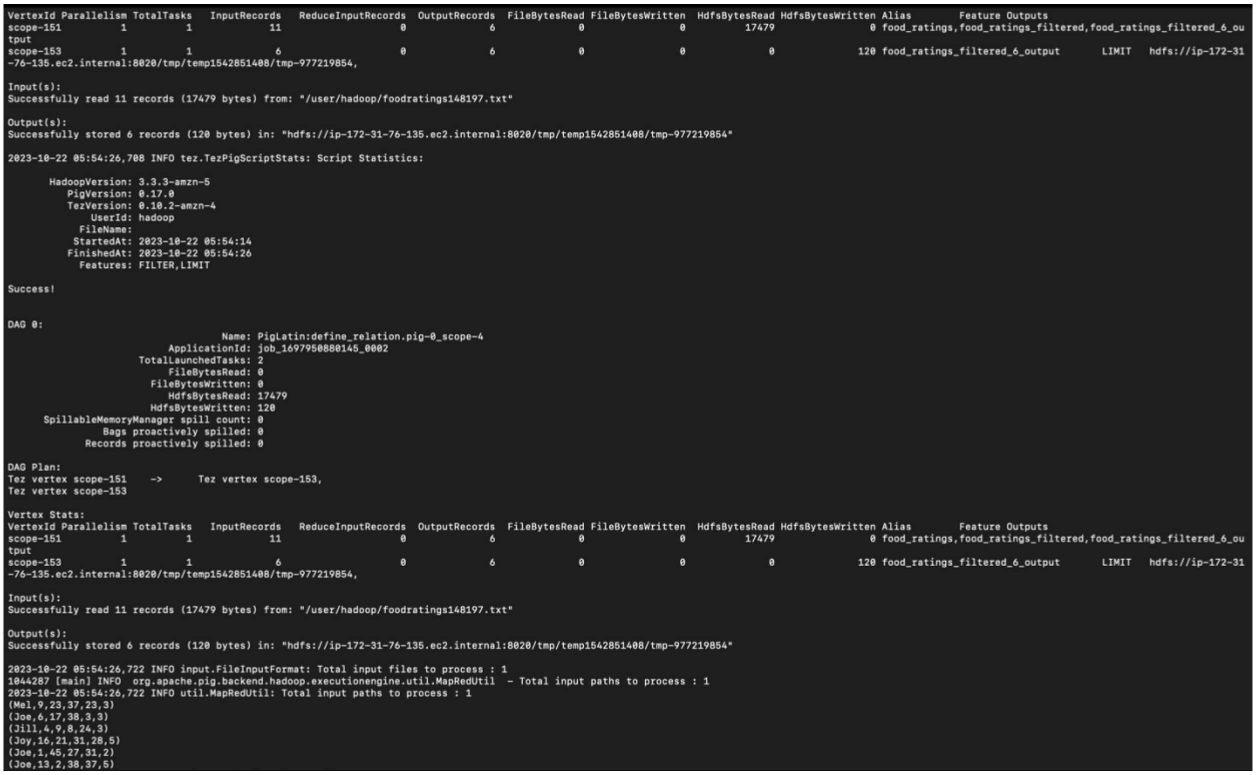


```
grunt> food_ratings_filtered = FILTER food_ratings BY (f1 < 20) AND (f3 > 5);
grunt>
grunt> food_ratings_filtered_6_output = LIMIT food_ratings_filtered 6;
grunt>
grunt> DUMP food_ratings_filtered_6_output;
```

Commands –

food_ratings_filtered = FILTER food_ratings BY (f1 < 20) AND (f3 > 5);
food_ratings_filtered_6_output = LIMIT food_ratings_filtered 6;
DUMP food_ratings_filtered_6_output;

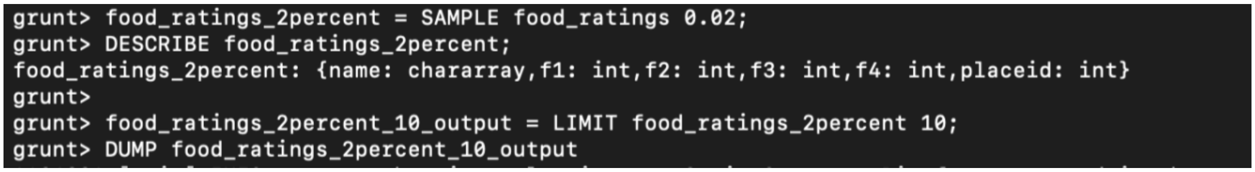
Write 6 records of this relation out to the console.



```
VertexId Parallelism TotalTasks   InputRecords   ReduceInputRecords  OutputRecords  FileBytesRead FileBytesWritten  HdfsBytesRead HdfsBytesWritten Alias        Feature Outputs
scope-151          1          1             11                    0              6              0                0          17479                0 food_ratings,food_ratings_filtered,food_ratings_filtered_6_ou
tput
scope-153          1          1              6                    0              6              0                0              0              120 food_ratings_filtered_6_output       LIMIT   hdfs://ip-172-31
-76-135.ec2.internal:8020/tmp/temp1542851408/tmp-977219854,

Input(s):
Successfully read 11 records (17479 bytes) from: "/user/hadoop/foodratings148197.txt"

Output(s):
Successfully stored 6 records (120 bytes) in: "hdfs://ip-172-31-76-135.ec2.internal:8020/tmp/temp1542851408/tmp-977219854"

2023-10-22 05:54:26,708 INFO tez.TezPigScriptStats: Script Statistics:

       HadoopVersion: 3.3.3-amzn-5
          PigVersion: 0.17.0
          TezVersion: 0.10.2-amzn-4
              UserId: hadoop
            FileName:
           StartedAt: 2023-10-22 05:54:14
          FinishedAt: 2023-10-22 05:54:26
            Features: FILTER,LIMIT

Success!


DAG 0:
                                    Name: PigLatin:define_relation.pig-0_scope-4
                           ApplicationId: job_1697950880145_0002
                       TotalLaunchedTasks: 2
                           FileBytesRead: 0
                        FileBytesWritten: 0
                           HdfsBytesRead: 17479
                        HdfsBytesWritten: 120
      SpillableMemoryManager spill count: 0
                Bags proactively spilled: 0
             Records proactively spilled: 0

DAG Plan:
Tez vertex scope-151    ->      Tez vertex scope-153,
Tez vertex scope-153

Vertex Stats:
VertexId Parallelism TotalTasks   InputRecords   ReduceInputRecords  OutputRecords  FileBytesRead FileBytesWritten  HdfsBytesRead HdfsBytesWritten Alias        Feature Outputs
scope-151          1          1             11                    0              6              0                0          17479                0 food_ratings,food_ratings_filtered,food_ratings_filtered_6_ou
tput
scope-153          1          1              6                    0              6              0                0              0              120 food_ratings_filtered_6_output       LIMIT   hdfs://ip-172-31
-76-135.ec2.internal:8020/tmp/temp1542851408/tmp-977219854,

Input(s):
Successfully read 11 records (17479 bytes) from: "/user/hadoop/foodratings148197.txt"

Output(s):
Successfully stored 6 records (120 bytes) in: "hdfs://ip-172-31-76-135.ec2.internal:8020/tmp/temp1542851408/tmp-977219854"

2023-10-22 05:54:26,722 INFO input.FileInputFormat: Total input files to process : 1
1044287 [main] INFO  org.apache.pig.backend.hadoop.executionengine.util.MapRedUtil  - Total input paths to process : 1
2023-10-22 05:54:26,722 INFO util.MapRedUtil: Total input paths to process : 1
(Mel,9,23,37,23,3)
(Joe,6,17,38,3,3)
(Jill,4,9,8,24,3)
(Joy,16,21,31,28,5)
(Joe,1,45,27,31,2)
(Joe,13,2,38,37,5)
```

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.

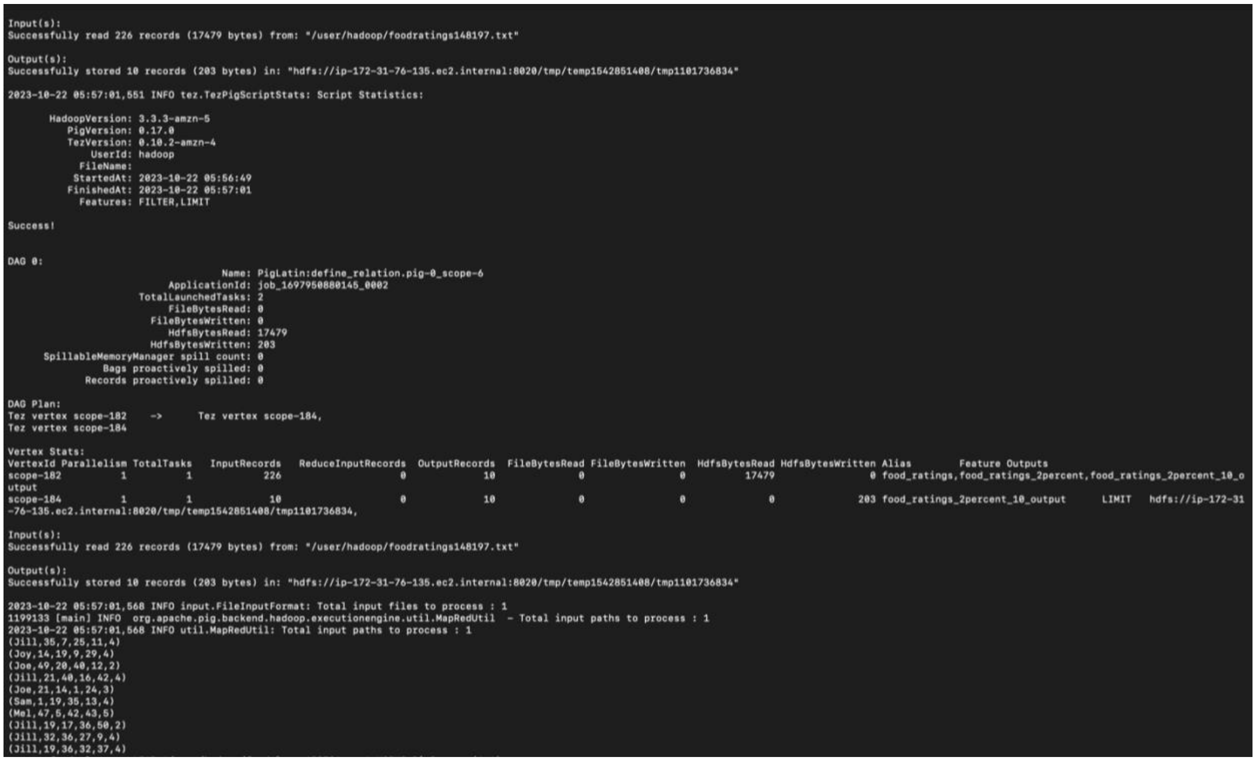


```
grunt> food_ratings_2percent = SAMPLE food_ratings 0.02;
grunt> DESCRIBE food_ratings_2percent;
food_ratings_2percent: {name: chararray,f1: int,f2: int,f3: int,f4: int,placeid: int}
grunt>
grunt> food_ratings_2percent_10_output = LIMIT food_ratings_2percent 10;
grunt> DUMP food_ratings_2percent_10_output
```

Commands –

food_ratings_2percent = SAMPLE food_ratings 0.02;
DESCRIBE food_ratings_2percent;
food_ratings_2percent_10_output = LIMIT food_ratings_2percent 10;
DUMP food_ratings_2percent_10_output

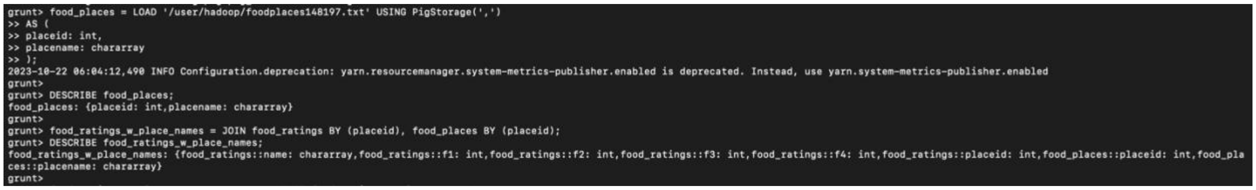
Write 10 of the records out to the console.



```
Input(s):
Successfully read 226 records (17479 bytes) from: "/user/hadoop/foodratings148197.txt"

Output(s):
Successfully stored 10 records (203 bytes) in: "hdfs://ip-172-31-76-135.ec2.internal:8020/tmp/temp1542851408/tmp1101736834"

2023-10-22 05:57:01,551 INFO tez.TezPigScriptStats: Script Statistics:

       HadoopVersion: 3.3.3-amzn-5
          PigVersion: 0.17.0
          TezVersion: 0.10.2-amzn-4
              UserId: hadoop
            FileName:
           StartedAt: 2023-10-22 05:56:49
          FinishedAt: 2023-10-22 05:57:01
            Features: FILTER,LIMIT

Success!


DAG 0:
                                    Name: PigLatin:define_relation.pig-0_scope-6
                           ApplicationId: job_1697950880145_0002
                      TotalLaunchedTasks: 2
                           FileBytesRead: 0
                        FileBytesWritten: 0
                           HdfsBytesRead: 17479
                        HdfsBytesWritten: 203
      SpillableMemoryManager spill count: 0
                Bags proactively spilled: 0
             Records proactively spilled: 0

DAG Plan:
Tez vertex scope-182    ->      Tez vertex scope-184,
Tez vertex scope-184

Vertex Stats:
VertexId Parallelism TotalTasks   InputRecords   ReduceInputRecords  OutputRecords  FileBytesRead FileBytesWritten  HdfsBytesRead HdfsBytesWritten Alias        Feature Outputs
scope-182          1          1            226                    0             10              0                0          17479                0 food_ratings,food_ratings_2percent,food_ratings_2percent_10_o
utput
scope-184          1          1             10                    0             10              0                0              0              203 food_ratings_2percent_10_output      LIMIT   hdfs://ip-172-31
-76-135.ec2.internal:8020/tmp/temp1542851408/tmp1101736834,

Input(s):
Successfully read 226 records (17479 bytes) from: "/user/hadoop/foodratings148197.txt"

Output(s):
Successfully stored 10 records (203 bytes) in: "hdfs://ip-172-31-76-135.ec2.internal:8020/tmp/temp1542851408/tmp1101736834"

2023-10-22 05:57:01,568 INFO input.FileInputFormat: Total input files to process : 1
1199133 [main] INFO  org.apache.pig.backend.hadoop.executionengine.util.MapRedUtil  - Total input paths to process : 1
2023-10-22 05:57:01,568 INFO util.MapRedUtil: Total input paths to process : 1
(Jill,35,7,25,11,4)
(Joy,14,19,9,29,4)
(Joe,49,20,40,12,2)
(Jill,21,40,16,42,4)
(Joe,21,14,1,24,3)
(Sam,1,19,35,13,4)
(Mel,47,5,42,43,5)
(Jill,19,17,36,50,2)
(Jill,32,36,27,9,4)
(Jill,19,36,32,37,4)
```

Exercise 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.

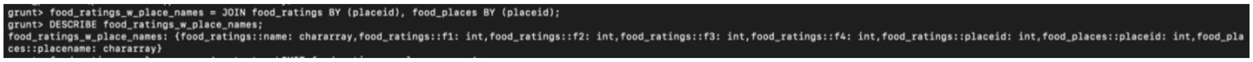


```
grunt> food_places = LOAD '/user/hadoop/foodplaces148197.txt' USING PigStorage(',')
>> AS (
>> placeid: int,
>> placename: chararray
>> );
2023-10-22 06:04:12,490 INFO Configuration.deprecation: yarn.resourcemanager.system-metrics-publisher.enabled is deprecated. Instead, use yarn.system-metrics-publisher.enabled
grunt>
grunt> DESCRIBE food_places;
food_places: {placeid: int,placename: chararray}
grunt>
grunt> food_ratings_w_place_names = JOIN food_ratings BY (placeid), food_places BY (placeid);
grunt> DESCRIBE food_ratings_w_place_names;
food_ratings_w_place_names: {food_ratings::name: chararray,food_ratings::f1: int,food_ratings::f2: int,food_ratings::f3: int,food_ratings::f4: int,food_ratings::placeid: int,food_places::placeid: int,food_pla
ces::placename: chararray}
grunt>
```

Commands –

```
food_places = LOAD '/user/hadoop/foodplaces148197.txt' USING PigStorage(',')

 AS (
 placeid: int,
placename: chararray
);
DESCRIBE food_places;
```

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.

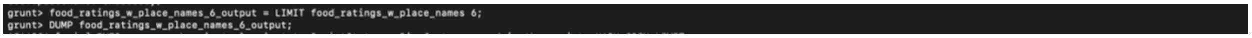


```
grunt> food_ratings_w_place_names = JOIN food_ratings BY (placeid), food_places BY (placeid);
grunt> DESCRIBE food_ratings_w_place_names;
food_ratings_w_place_names: {food_ratings::name: chararray,food_ratings::f1: int,food_ratings::f2: int,food_ratings::f3: int,food_ratings::f4: int,food_ratings::placeid: int,food_places::placeid: int,food_pla
ces::placename: chararray}
```

Commands –

```
food_ratings_w_place_names = JOIN food_ratings BY (placeid), food_places BY (placeid);
DESCRIBE food_ratings_w_place_names;
```

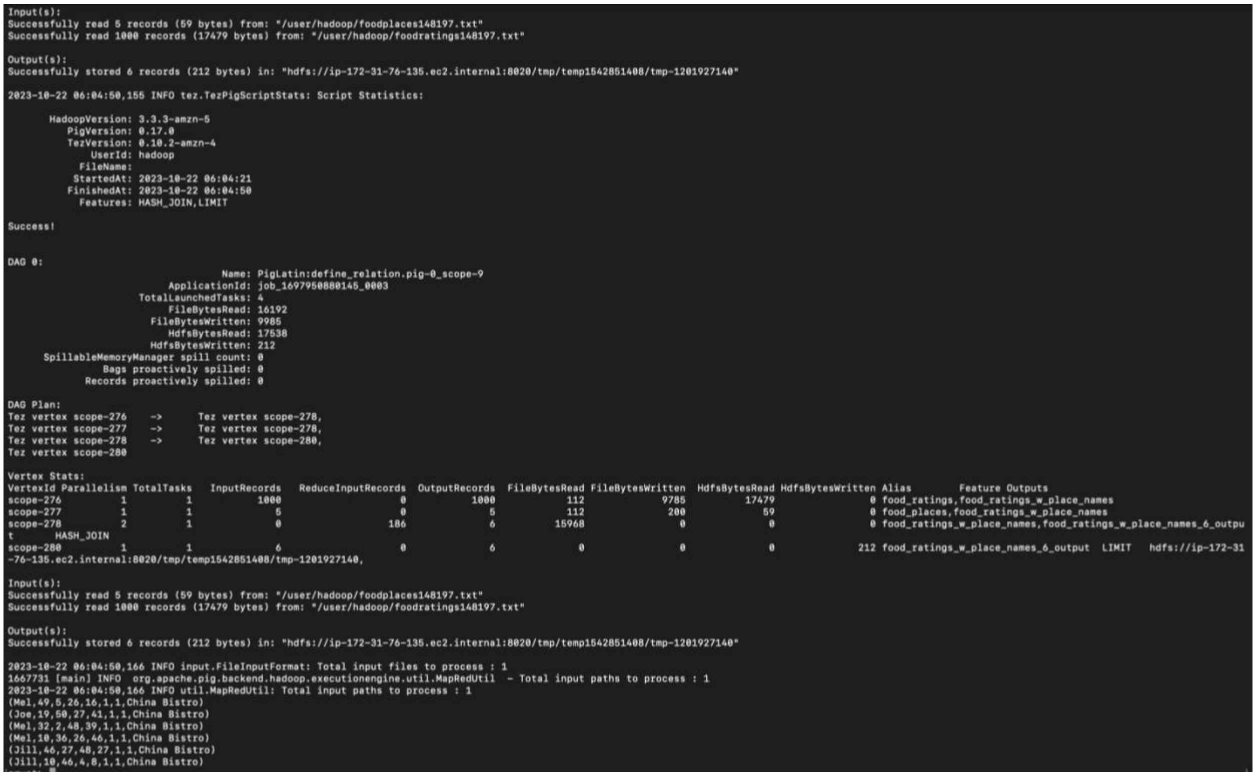
Write 6 records of this relation out to the console.



```
grunt> food_ratings_w_place_names_6_output = LIMIT food_ratings_w_place_names 6;
grunt> DUMP food_ratings_w_place_names_6_output;
```

Commands –

food_ratings_w_place_names_6_output = LIMIT food_ratings_w_place_names 6;
DUMP food_ratings_w_place_names_6_output;



```
Input(s):
Successfully read 5 records (59 bytes) from: "/user/hadoop/foodplaces148197.txt"
Successfully read 1000 records (17479 bytes) from: "/user/hadoop/foodratings148197.txt"

Output(s):
Successfully stored 6 records (212 bytes) in: "hdfs://ip-172-31-76-135.ec2.internal:8020/tmp/temp1542851408/tmp-1201927140"

2023-10-22 06:04:50,155 INFO tez.TezPigScriptStats: Script Statistics:

       HadoopVersion: 3.3.3-amzn-5
          PigVersion: 0.17.0
          TezVersion: 0.10.2-amzn-4
              UserId: hadoop
            FileName:
           StartedAt: 2023-10-22 06:04:21
          FinishedAt: 2023-10-22 06:04:50
            Features: HASH_JOIN,LIMIT

Success!


DAG 0:
                                    Name: PigLatin:define_relation.pig-0_scope-9
                           ApplicationId: job_1697950880145_0003
                      TotalLaunchedTasks: 4
                           FileBytesRead: 16192
                        FileBytesWritten: 9985
                           HdfsBytesRead: 17538
                        HdfsBytesWritten: 212
      SpillableMemoryManager spill count: 0
                Bags proactively spilled: 0
             Records proactively spilled: 0

DAG Plan:
Tez vertex scope-276    ->      Tez vertex scope-278,
Tez vertex scope-277    ->      Tez vertex scope-278,
Tez vertex scope-278    ->      Tez vertex scope-280,
Tez vertex scope-280

Vertex Stats:
VertexId Parallelism TotalTasks   InputRecords   ReduceInputRecords  OutputRecords  FileBytesRead FileBytesWritten  HdfsBytesRead HdfsBytesWritten Alias        Feature Outputs
scope-276          1          1           1000                    0           1000            112             9785          17479                0 food_ratings,food_ratings_w_place_names
scope-277          1          1              5                    0              5            112              200             59                0 food_places,food_ratings_w_place_names
scope-278          2          1              0                  186              6          15968                0              0                0 food_ratings_w_place_names,food_ratings_w_place_names_6_outpu
t       HASH_JOIN
scope-280          1          1              6                    0              6              0                0              0              212 food_ratings_w_place_names_6_output  LIMIT   hdfs://ip-172-31
-76-135.ec2.internal:8020/tmp/temp1542851408/tmp-1201927140,

Input(s):
Successfully read 5 records (59 bytes) from: "/user/hadoop/foodplaces148197.txt"
Successfully read 1000 records (17479 bytes) from: "/user/hadoop/foodratings148197.txt"

Output(s):
Successfully stored 6 records (212 bytes) in: "hdfs://ip-172-31-76-135.ec2.internal:8020/tmp/temp1542851408/tmp-1201927140"

2023-10-22 06:04:50,166 INFO input.FileInputFormat: Total input files to process : 1
1667731 [main] INFO  org.apache.pig.backend.hadoop.executionengine.util.MapRedUtil  - Total input paths to process : 1
2023-10-22 06:04:50,166 INFO util.MapRedUtil: Total input paths to process : 1
(Mel,49,5,26,16,1,1,China Bistro)
(Joe,19,50,27,41,1,1,China Bistro)
(Mel,32,2,48,39,1,1,China Bistro)
(Mel,10,36,26,46,1,1,China Bistro)
(Jill,46,27,48,27,1,1,China Bistro)
(Jill,10,46,4,8,1,1,China Bistro)
```

Exercise 7)

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

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

Answer: LIMIT

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

A. SAMPLE
B. FILTER
C. DISTINCT
D. SPLIT

Answer: DISTINCT

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?

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)

Answer: (f1: STRING, f2: INT, f3: INT, 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)

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;

Answer: relB = FOREACH relA GENERATE $0, f3;

V. Pig Latin is a ________ language. Select the best choice to fill in the blank.

A. functional

B. data flow

C. procedural

D. declarative

ANS: Pig Latin is a DATA FLOW language.

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

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

Answer: relB = FILTER relA by $0 < 20