

- a. **Top 5 employees (employee id and employee name) with highest rating. (In case two employees have same rating, employee with name coming first in dictionary should get preference)**

Below is the script used to derive top 5 employees with highest rating(names ordered in alphabetic manner)-

Pig -x local;

emp_details = LOAD '/home/acadgild/pig/employee_details.txt' USING PigStorage(',') AS (emp_id:int, name:chararray, salary:int, rating:int);

grpd = GROUP emp_details BY rating;

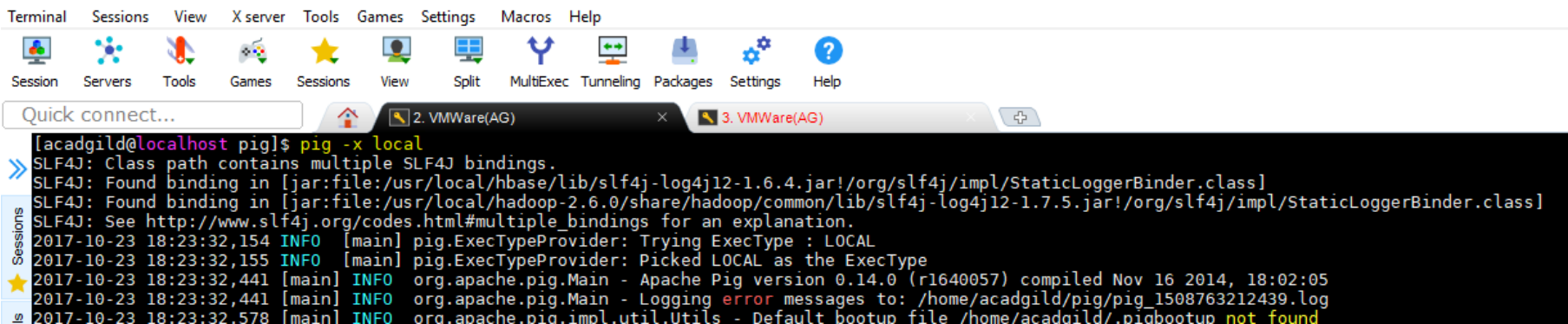
```
top5 = FOREACH grpd {  
    sorted = ORDER emp_details by name ASC;  
  
    top1 = limit sorted 1;  
  
    generate FLATTEN(top1);  
};
```

grpd2 = FOREACH top5 GENERATE emp_id, name;

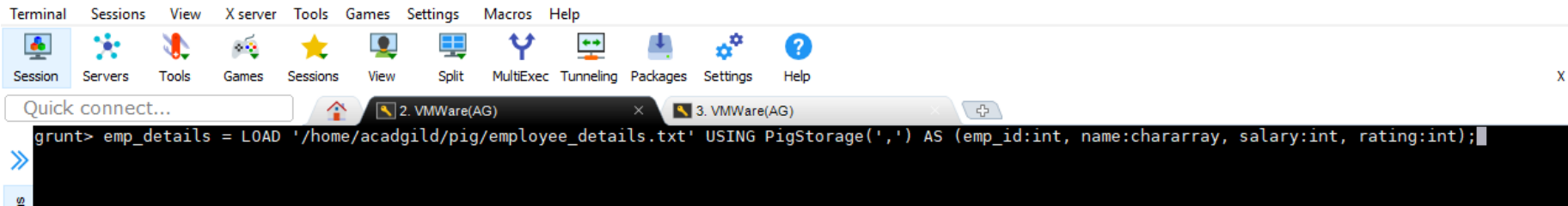
dump grpd2;

Each and every relation has been explained below with its immediate corresponding output-

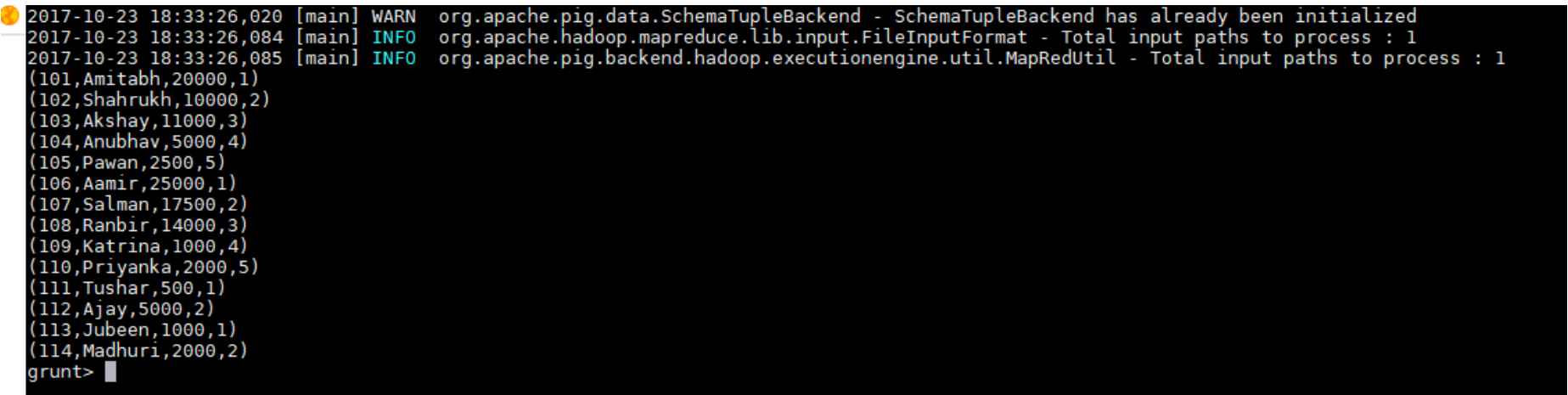
Start pig in local mode-



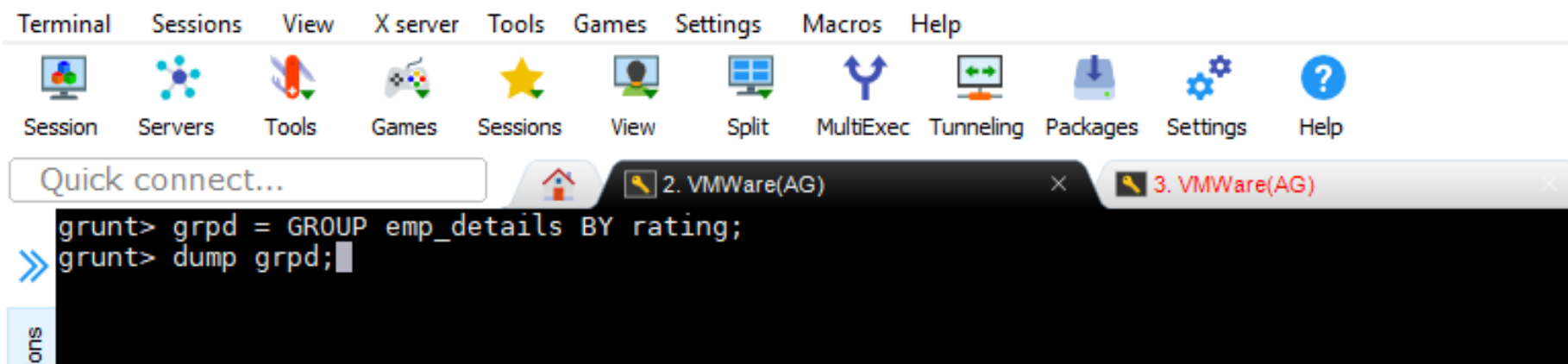
Load the emp_details.txt file using PigStorage



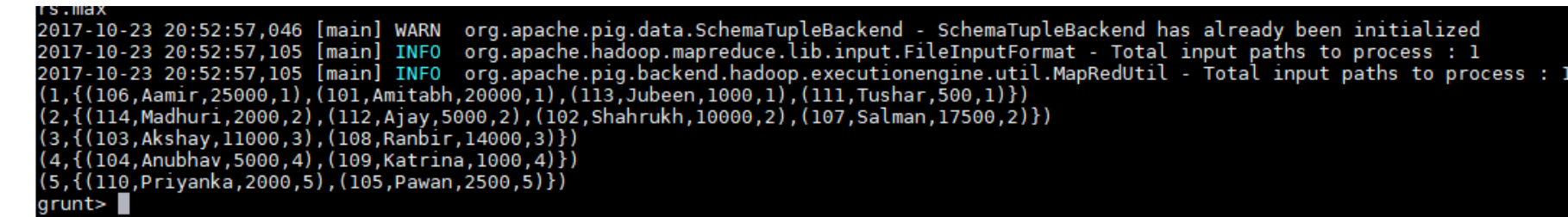
Contents of file-



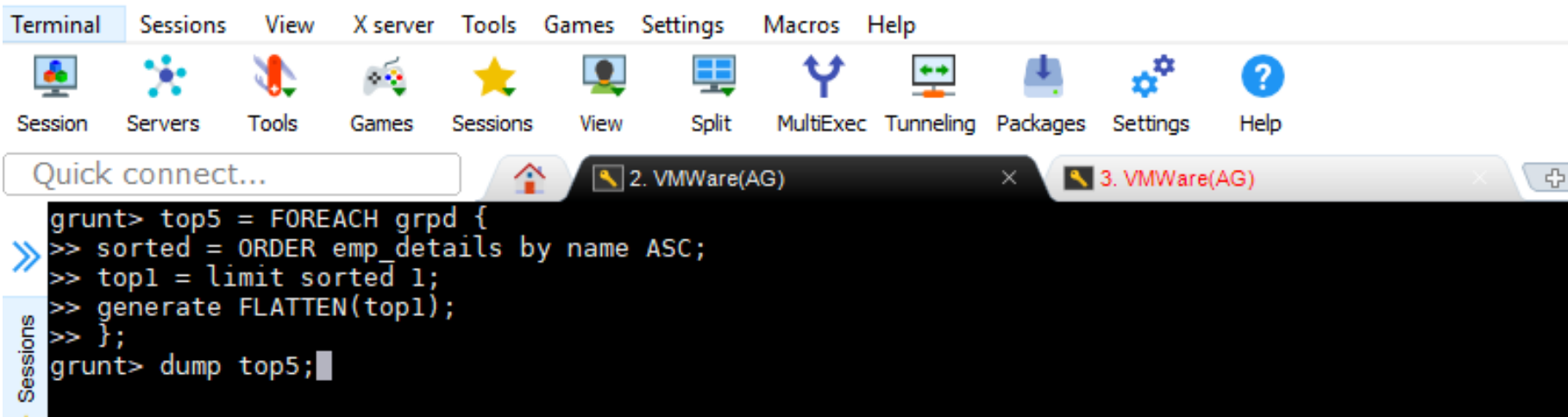
Group the data by rating-



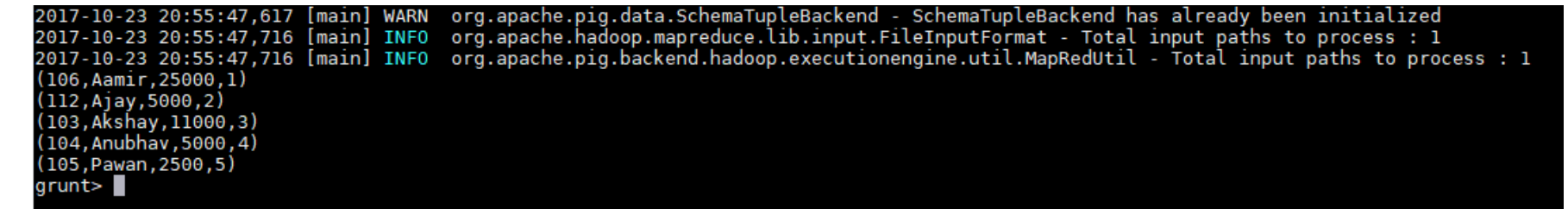
Result of GROUP-



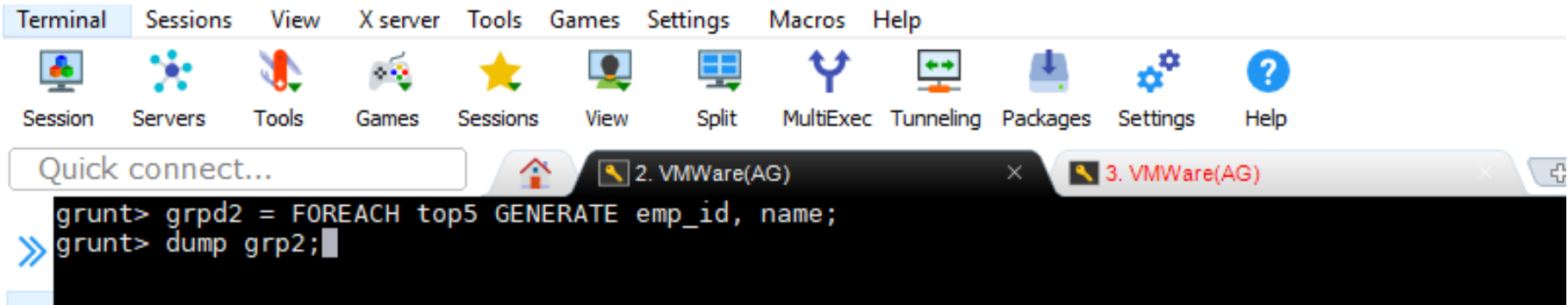
Below script takes out the tuple from the bag arranged with name in alphabetic order-



Dump results-



Extract emp_id and name as per question-



Final O/P-

```
2017-10-23 20:57:19,863 [main] WARN org.apache.pig.data.SchemaTupleBackend - SchemaTupleBackend has already been initialized
2017-10-23 20:57:19,927 [main] INFO org.apache.hadoop.mapreduce.lib.input.FileInputFormat - Total input paths to process : 1
2017-10-23 20:57:19,927 [main] INFO org.apache.pig.backend.hadoop.executionengine.util.MapRedUtil - Total input paths to process : 1
(106,Aamir)
(112,Ajay)
(103,Akshay)
(104,Anubhav)
(105,Pawan)
grunt>
```

b. Top 3 employees (employee id and employee name) with highest salary, whose employee id is an odd number. (In case two employees have same salary, employee with name coming first in dictionary should get preference)

Below is the script used to finalize the result-

```
Pig -x local;

emp_details = LOAD '/home/acadgild/pig/employee_details.txt' USING PigStorage(',') AS (emp_id:int, name:chararray, salary:int, rating:int);

odd_emp = FILTER emp_details BY emp_id % 2 != 0;

order_sal = ORDER odd_emp BY salary DESC;

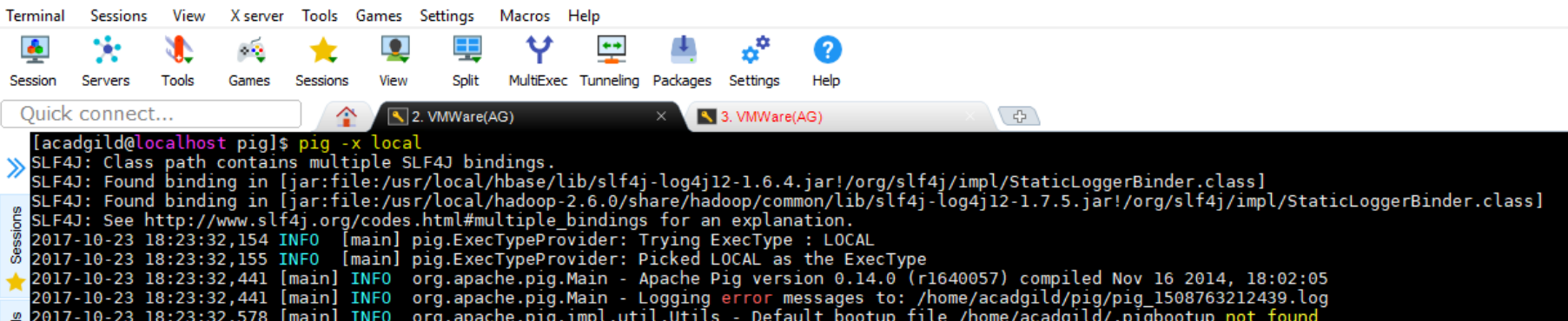
top3 = LIMIT order_sal 3;

final_res = FOREACH top3 GENERATE (emp_id, name);

dump final_res;
```

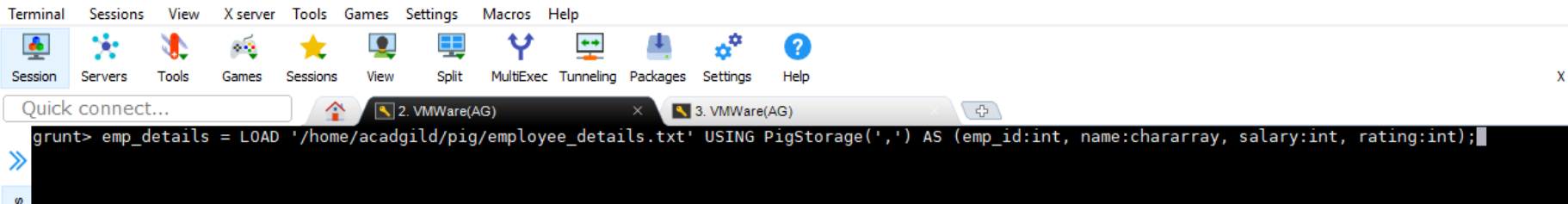
Each and every relation has been explained below with its immediate corresponding output-

Start pig in local mode-



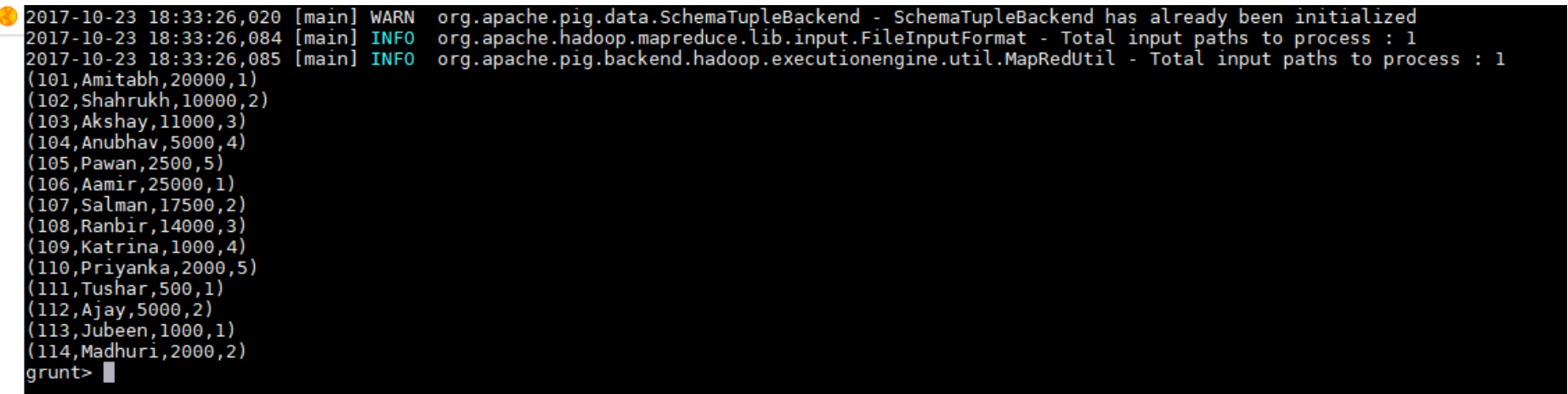
```
[acadgild@localhost pig]$ pig -x local
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/usr/local/hbase/lib/slf4j-log4j12-1.6.4.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/usr/local/hadoop-2.6.0/share/hadoop/common/lib/slf4j-log4j12-1.7.5.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
2017-10-23 18:23:32,154 INFO [main] pig.ExecTypeProvider: Trying ExecType : LOCAL
2017-10-23 18:23:32,155 INFO [main] pig.ExecTypeProvider: Picked LOCAL as the ExecType
2017-10-23 18:23:32,441 INFO [main] org.apache.pig.Main - Apache Pig version 0.14.0 (r1640057) compiled Nov 16 2014, 18:02:05
2017-10-23 18:23:32,441 INFO [main] org.apache.pig.Main - Logging error messages to: /home/acadgild/pig/pig_1508763212439.log
2017-10-23 18:23:32,578 INFO [main] org.apache.pig.impl.util.Utils - Default bootstrap file /home/acadgild/.pigbootstrap not found
```

Load the emp_details.txt file using PigStorage



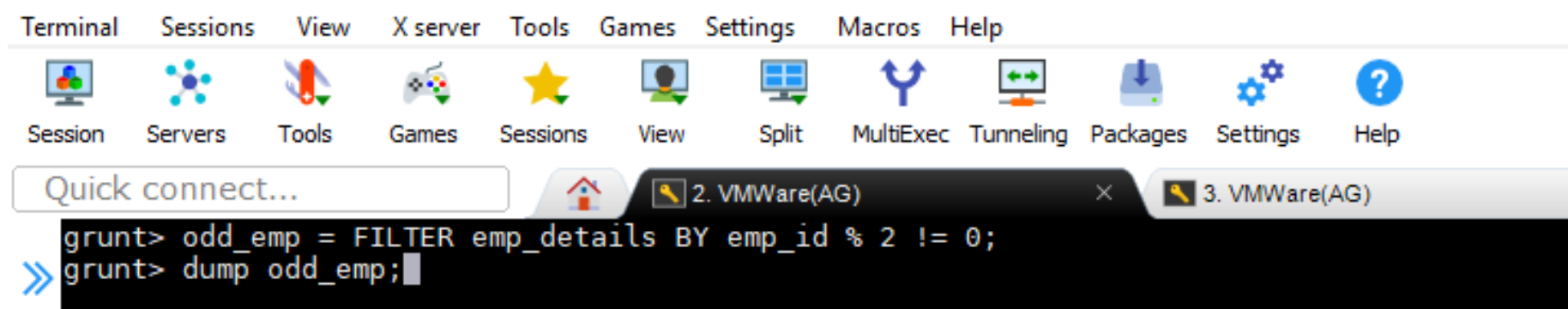
```
grunt> emp_details = LOAD '/home/acadgild/pig/employee_details.txt' USING PigStorage(',') AS (emp_id:int, name:chararray, salary:int, rating:int);
```

Contents of file-

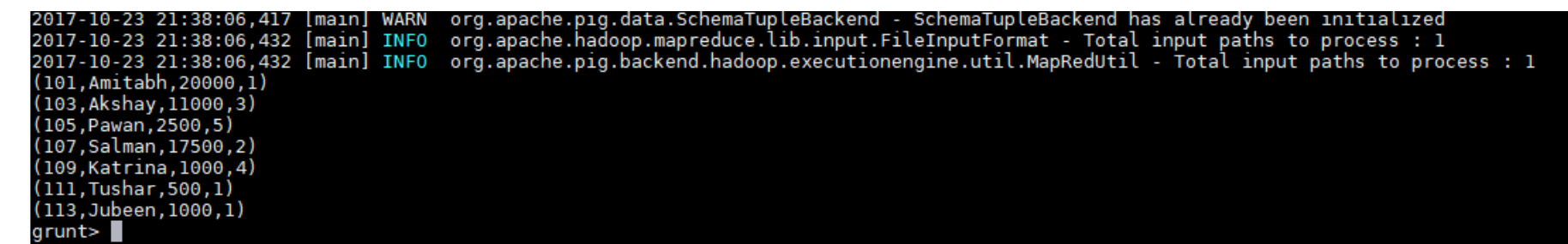


```
2017-10-23 18:33:26,020 [main] WARN org.apache.pig.data.SchemaTupleBackend - SchemaTupleBackend has already been initialized
2017-10-23 18:33:26,084 [main] INFO org.apache.hadoop.mapreduce.lib.input.FileInputFormat - Total input paths to process : 1
2017-10-23 18:33:26,085 [main] INFO org.apache.pig.backend.hadoop.executionengine.util.MapRedUtil - Total input paths to process : 1
(101,Amitabh,20000,1)
(102,Shahrukh,10000,2)
(103,Akshay,11000,3)
(104,Anubhav,5000,4)
(105,Pawan,2500,5)
(106,Aamir,25000,1)
(107,Salman,17500,2)
(108,Ranbir,14000,3)
(109,Katrina,1000,4)
(110,Priyanka,2000,5)
(111,Tushar,500,1)
(112,Ajay,5000,2)
(113,Jubeen,1000,1)
(114,Madhuri,2000,2)
grunt>
```

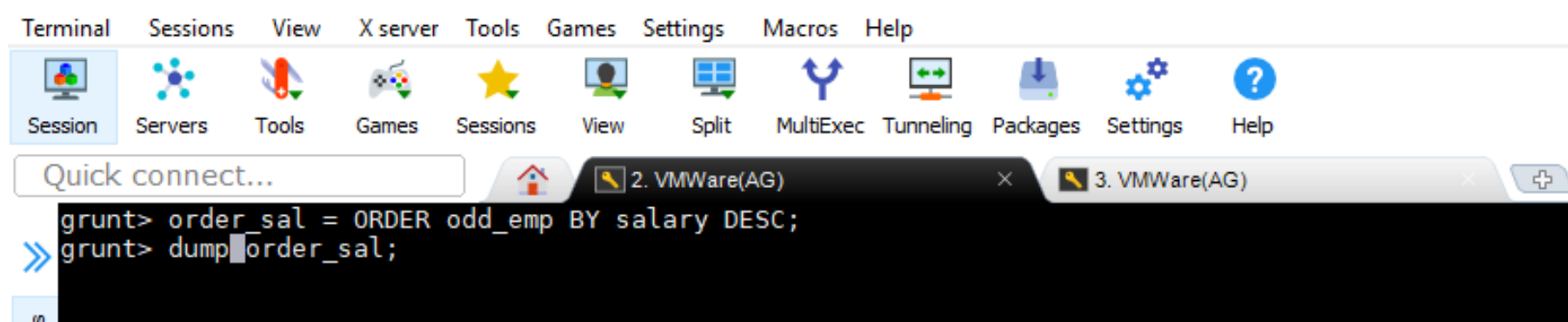
Filtering out the employees with odd emp_id-



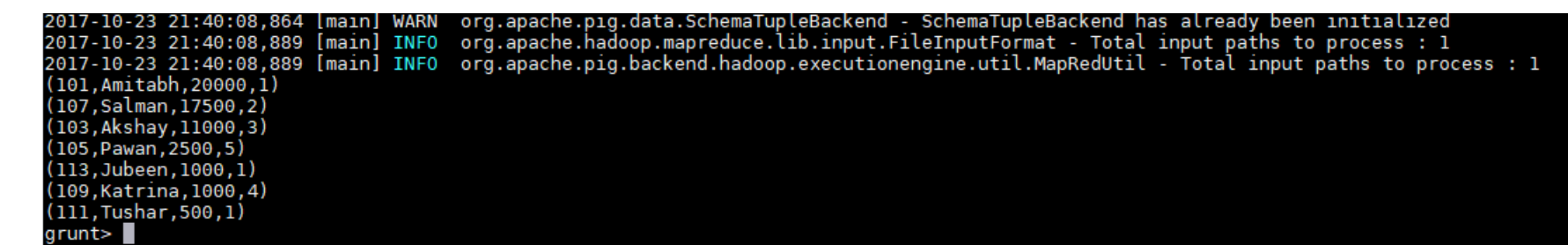
Records with odd emp_ids-



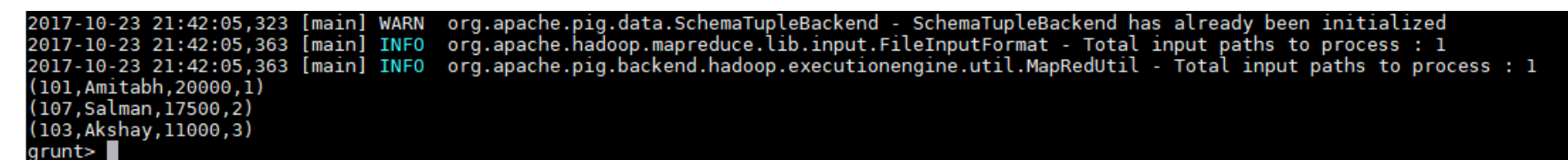
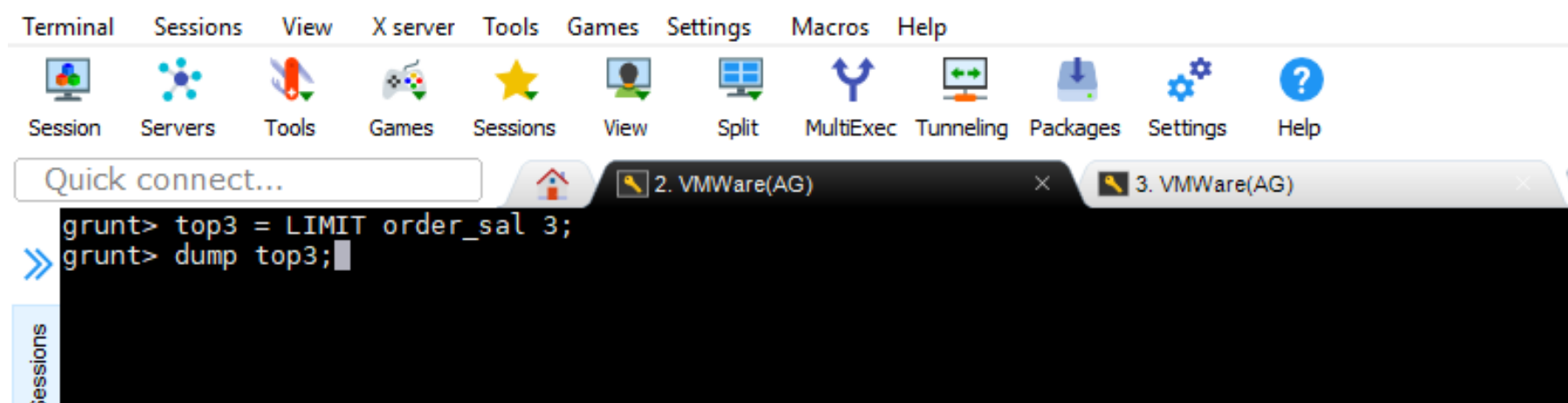
Ordering the salary in descending manner-



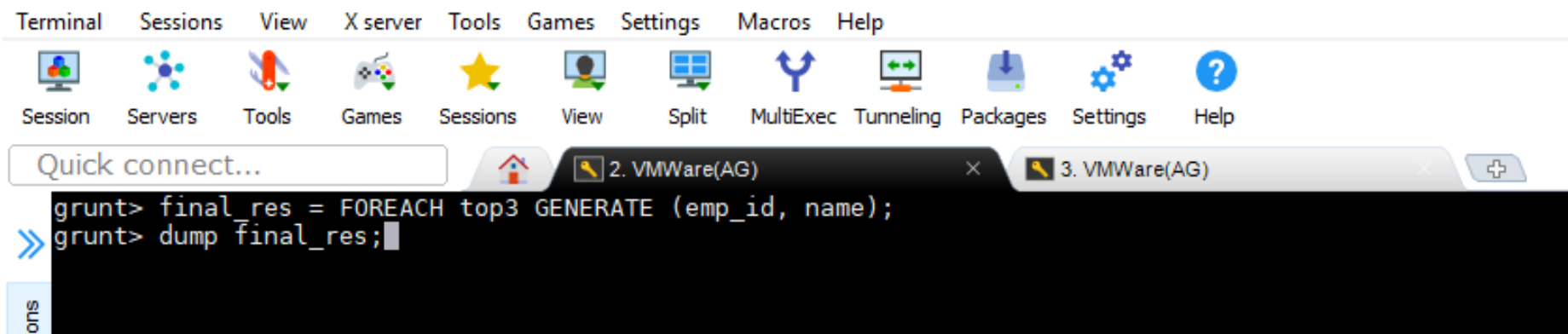
Records with descending order of salary-



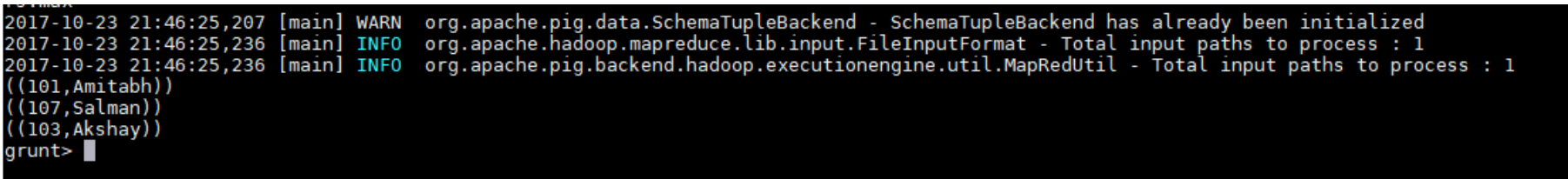
Extracting top 3 rows for top 3 employees with odd emp_id-



Generating emp_id and emp_name as per question-



Final output-



- c. Employee (employee id and employee name) with maximum expense (In case two employees have same expense, employee with name coming first in dictionary should get preference)

Below is the script used to finalize the result-

Pig -x local;

emp_details = LOAD '/home/acadgild/pig/employee_details.txt' USING PigStorage(',') AS (emp_id:int, name:chararray, salary:int, rating:int);

emp_expense = LOAD '/home/acadgild/pig/employee_expenses.txt' USING PigStorage('\t') AS (emp_id:int, expense:int);

joined_data = JOIN emp_details by emp_id, emp_expense by emp_id;

ordered_data = ORDER joined_data BY expense DESC, name ASC;

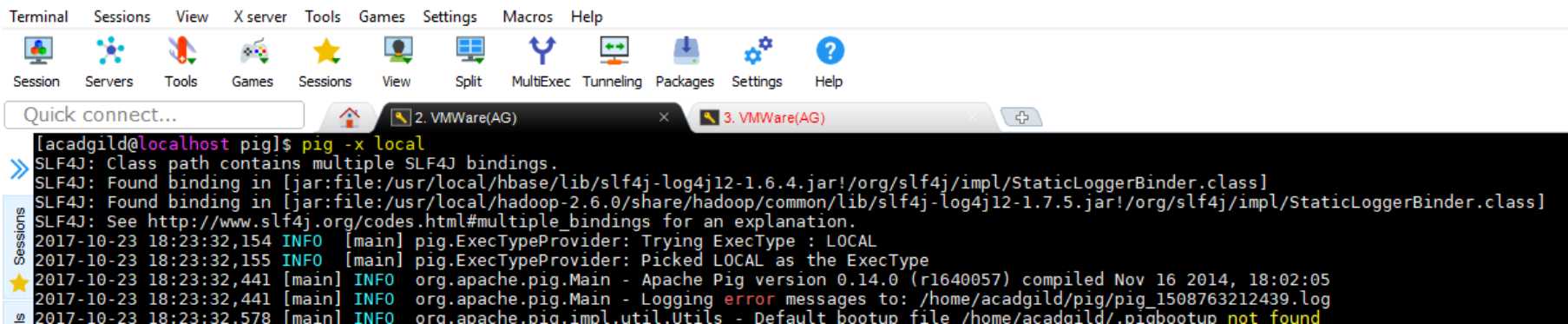
extract_id_name = FOREACH ordered_data GENERATE \$0 AS emp_id, \$1 AS name;

Final_result = LIMIT extract_id_name 1;

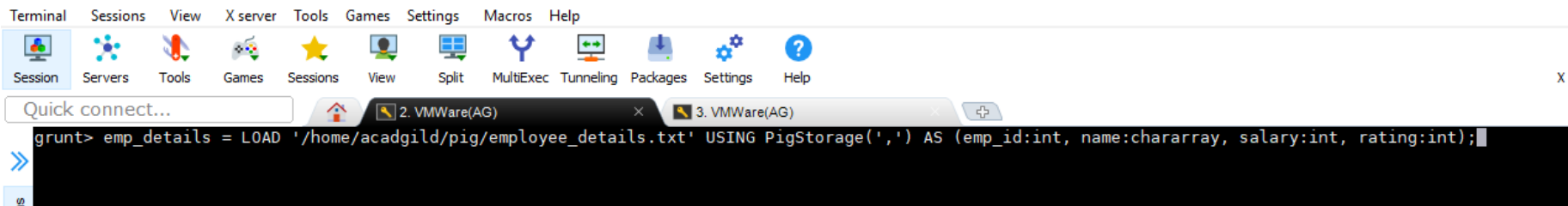
dump Final_result;

Each and every relation has been explained below with its immediate corresponding output-

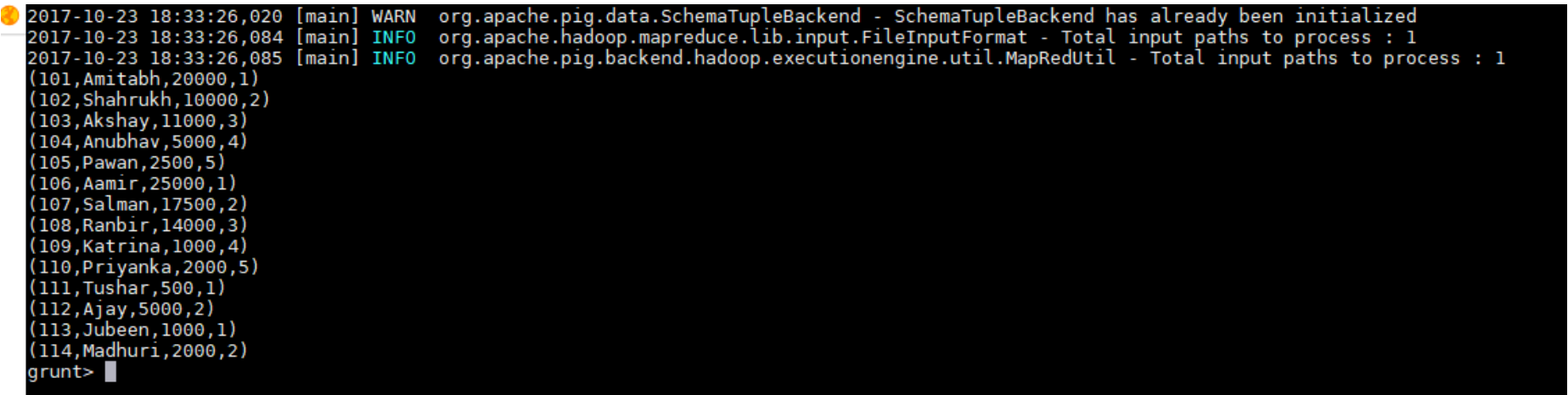
Start pig in local mode-



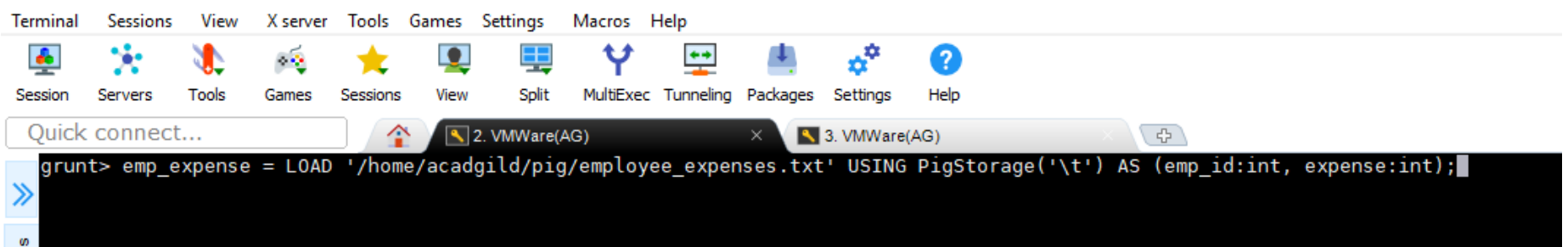
Load the emp_details.txt file using PigStorage



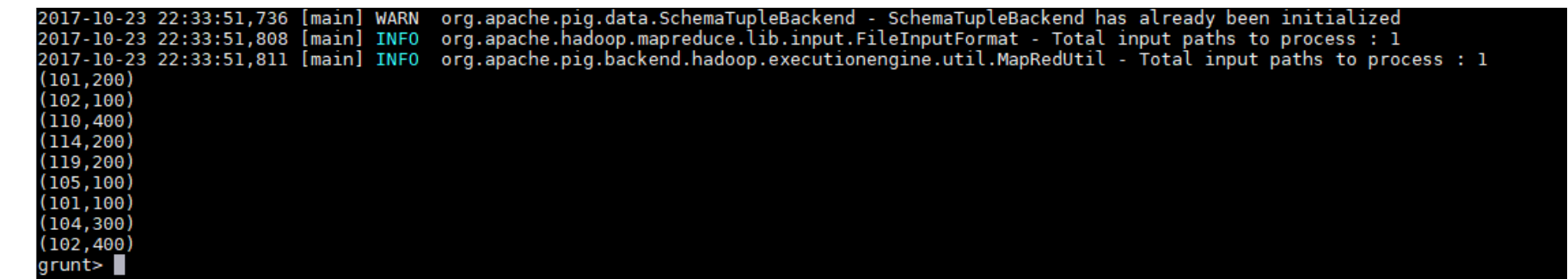
Contents of file-



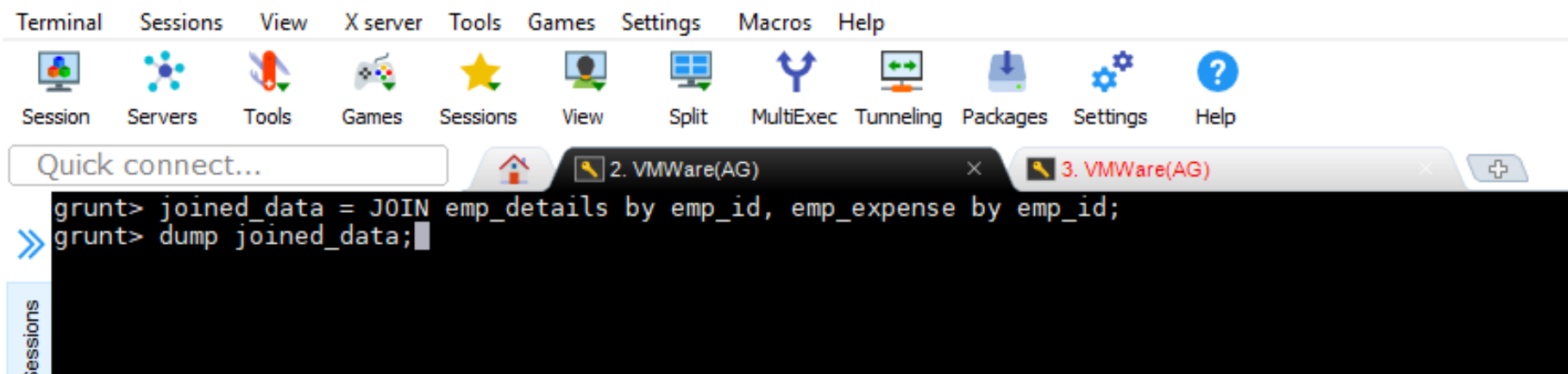
Load the emp_expense.txt file using PigStorage



Contents of file-



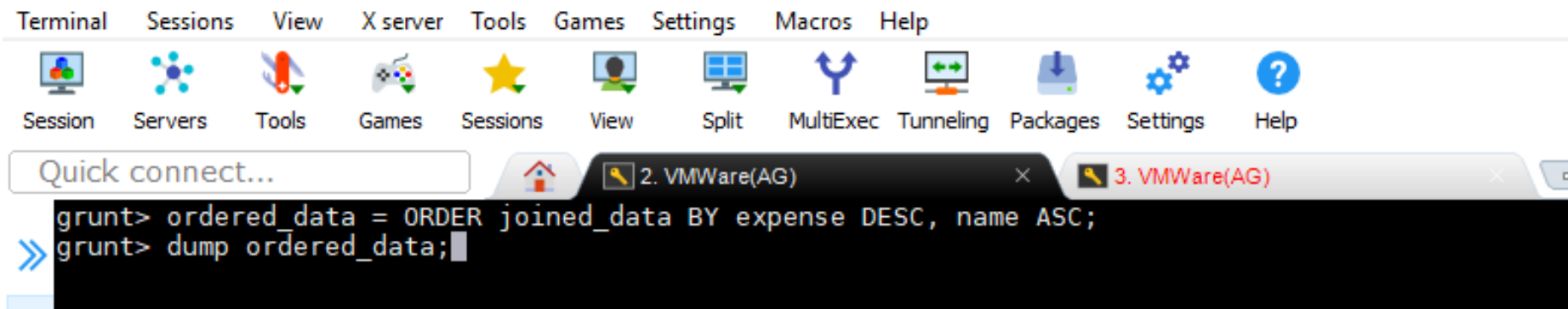
Join both files using JOIN relation-



Result after JOIN

```
2017-10-23 22:45:06,477 [main] WARN org.apache.pig.data.SchemaTupleBackend - SchemaTupleBackend has already been initialized
2017-10-23 22:45:06,541 [main] INFO org.apache.hadoop.mapreduce.lib.input.FileInputFormat - Total input paths to process : 1
2017-10-23 22:45:06,542 [main] INFO org.apache.pig.backend.hadoop.executionengine.util.MapRedUtil - Total input paths to process : 1
(101,Amitabh,20000,1,101,100)
(101,Amitabh,20000,1,101,200)
(102,Shahrukh,10000,2,102,400)
(102,Shahrukh,10000,2,102,100)
(104,Anubhav,5000,4,104,300)
(105,Pawan,2500,5,105,100)
(110,Priyanka,2000,5,110,400)
(114,Madhuri,2000,2,114,200)
grunt>
```

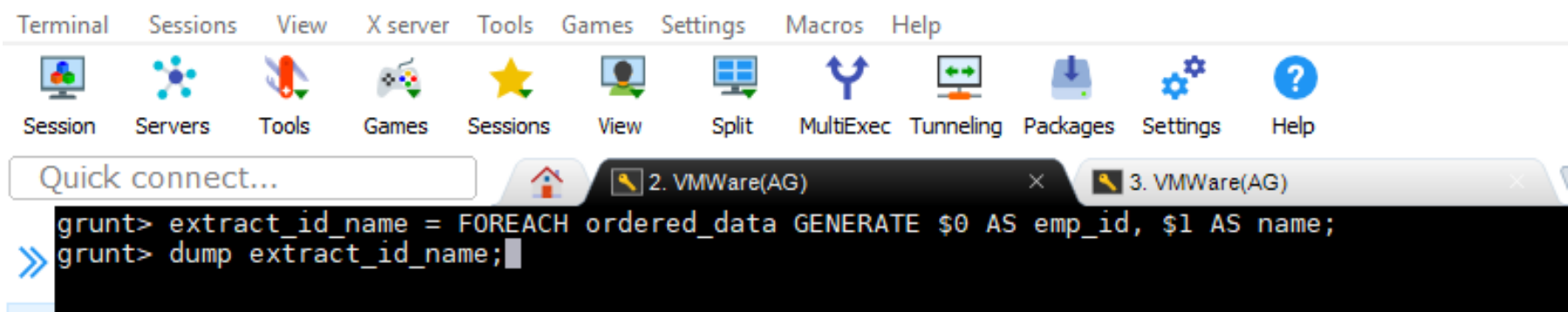
ORDER the JOIN output with expense in Descending order and name in Ascending order to take the name of employee in alphabetic manner with highest expense-



Result

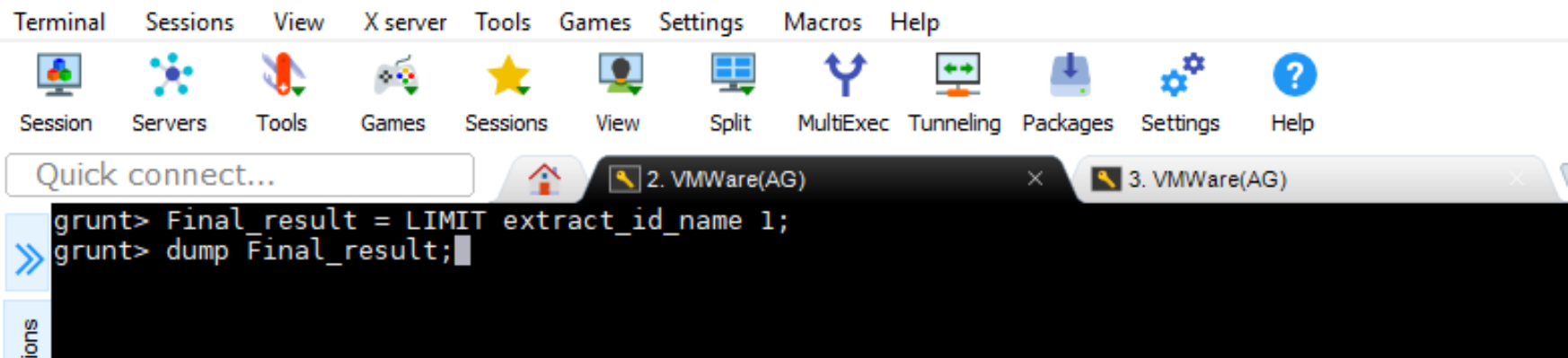
```
2017-10-23 23:52:58,966 [main] WARN org.apache.pig.data.SchemaTupleBackend - SchemaTupleBackend has already been initialized
2017-10-23 23:52:59,020 [main] INFO org.apache.hadoop.mapreduce.lib.input.FileInputFormat - Total input paths to process : 1
2017-10-23 23:52:59,020 [main] INFO org.apache.pig.backend.hadoop.executionengine.util.MapRedUtil - Total input paths to process : 1
(110,Priyanka,2000,5,110,400)
(102,Shahrukh,10000,2,102,400)
(104,Anubhav,5000,4,104,300)
(101,Amitabh,20000,1,101,200)
(114,Madhuri,2000,2,114,200)
(101,Amitabh,20000,1,101,100)
(105,Pawan,2500,5,105,100)
(102,Shahrukh,10000,2,102,100)
grunt>
```

Extract only emp_id and emp_name as required -



```
2017-10-24 15:16:55,855 [main] WARN org.apache.pig.data.SchemaTuppleBackend - SchemaTuppleBackend has already been initialized
2017-10-24 15:16:55,887 [main] INFO org.apache.hadoop.mapreduce.lib.input.FileInputFormat - Total input paths to process : 1
2017-10-24 15:16:55,887 [main] INFO org.apache.pig.backend.hadoop.executionengine.util.MapRedUtil - Total input paths to process : 1
(110,Priyanka)
(102,Shahrukh)
(104,Anubhav)
(101,Amitabh)
(114,Madhuri)
(101,Amitabh)
(105,Pawan)
(102,Shahrukh)
grunt>
```

Select the topmost row using LIMIT-



Final O/P

```
2017-10-24 15:22:14,238 [main] WARN org.apache.pig.data.SchemaTuppleBackend - SchemaTuppleBackend has already been initialized
2017-10-24 15:22:14,265 [main] INFO org.apache.hadoop.mapreduce.lib.input.FileInputFormat - Total input paths to process : 1
2017-10-24 15:22:14,265 [main] INFO org.apache.pig.backend.hadoop.executionengine.util.MapRedUtil - Total input paths to process : 1
(110,Priyanka)
grunt>
```

d. List of employees (employee id and employee name) having entries in employee expenses file.

Below is the script used to finalize the result-

```
Pig -x local;

emp_details = LOAD '/home/acadgild/pig/employee_details.txt' USING PigStorage(',') AS (emp_id:int, name:chararray, salary:int, rating:int);

emp_expense = LOAD '/home/acadgild/pig/employee_expenses.txt' USING PigStorage('\t') AS (emp_id:int, expense:int);

joined_data = JOIN emp_details by emp_id, emp_expense by emp_id;

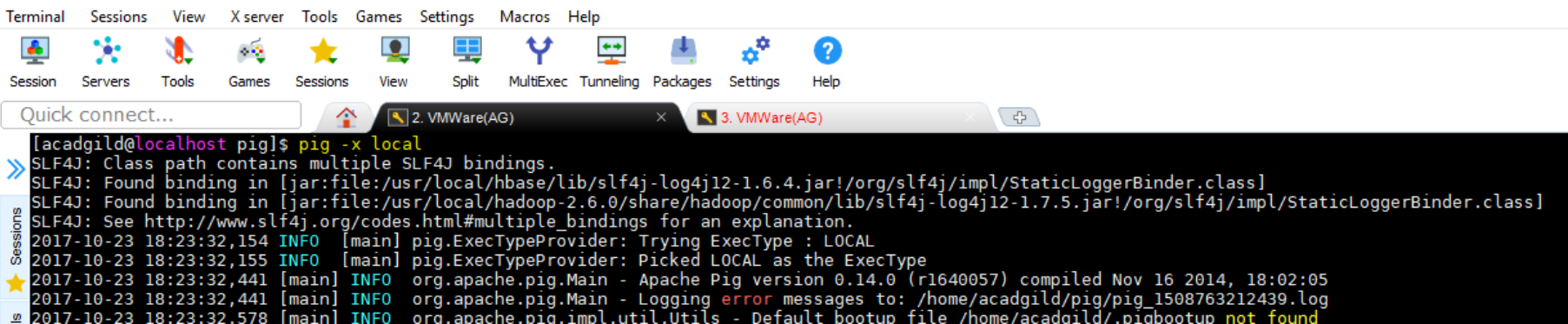
extract_id_name = FOREACH joined_data GENERATE $0 AS emp_id, $1 AS name;

Final_result = DISTINCT extract_id_name;

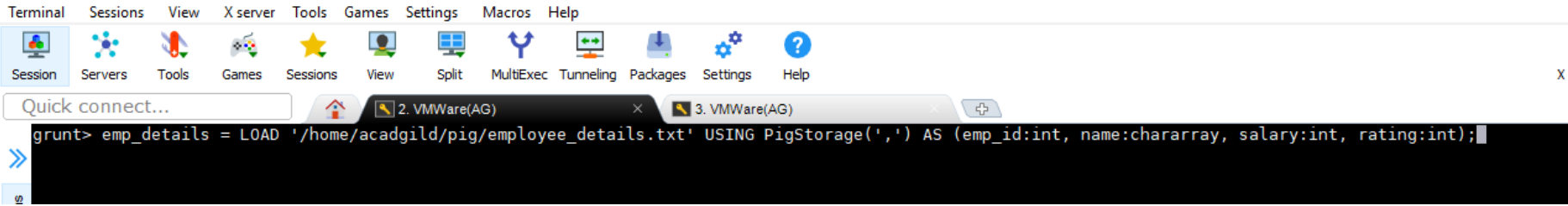
dump Final_result;
```

Each and every relation has been explained below with its immediate corresponding output-

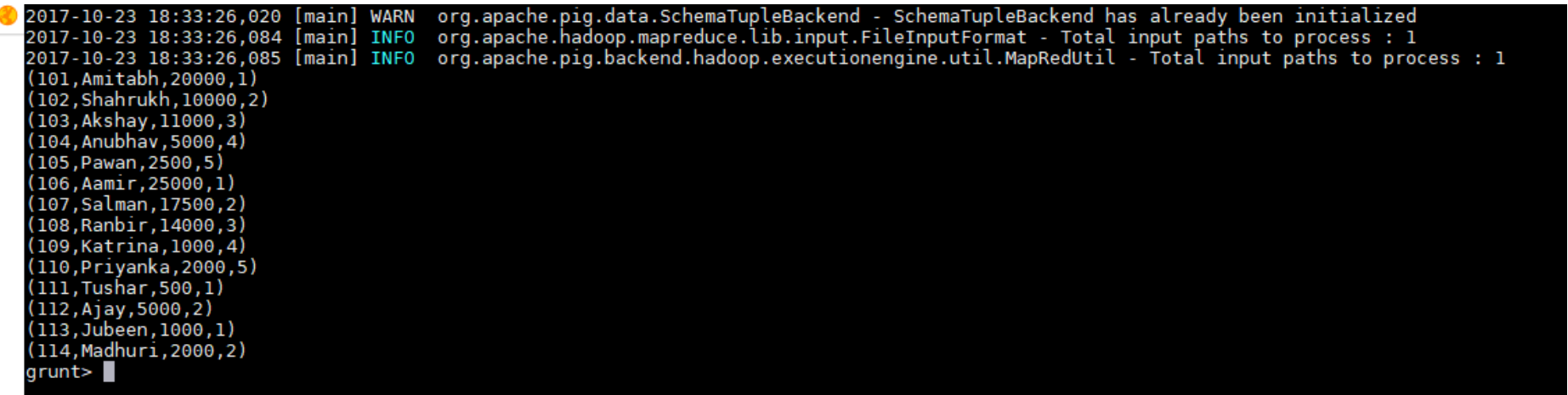
Start pig in local mode-



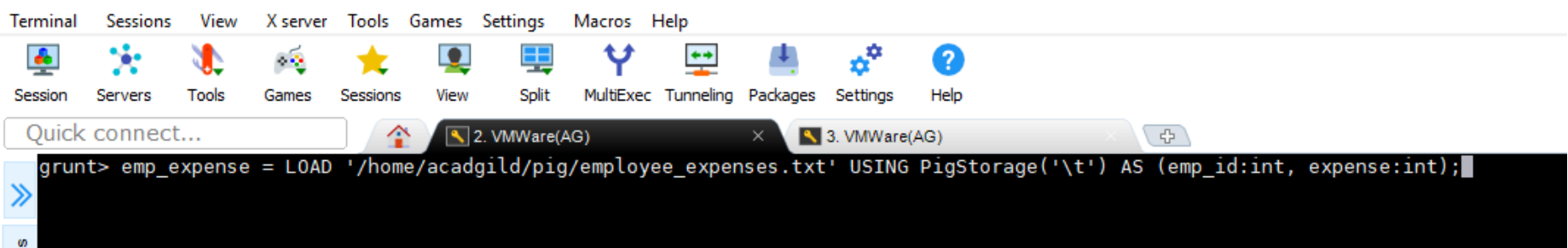
Load the emp_details.txt file using PigStorage



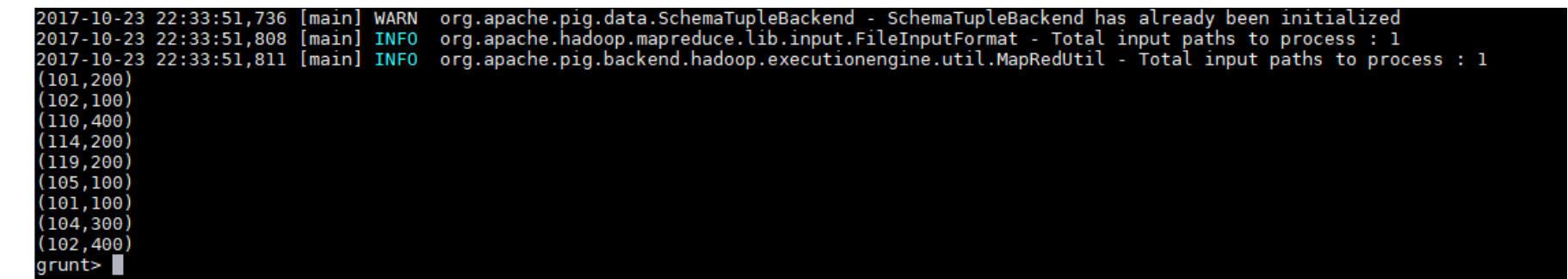
Contents of file-



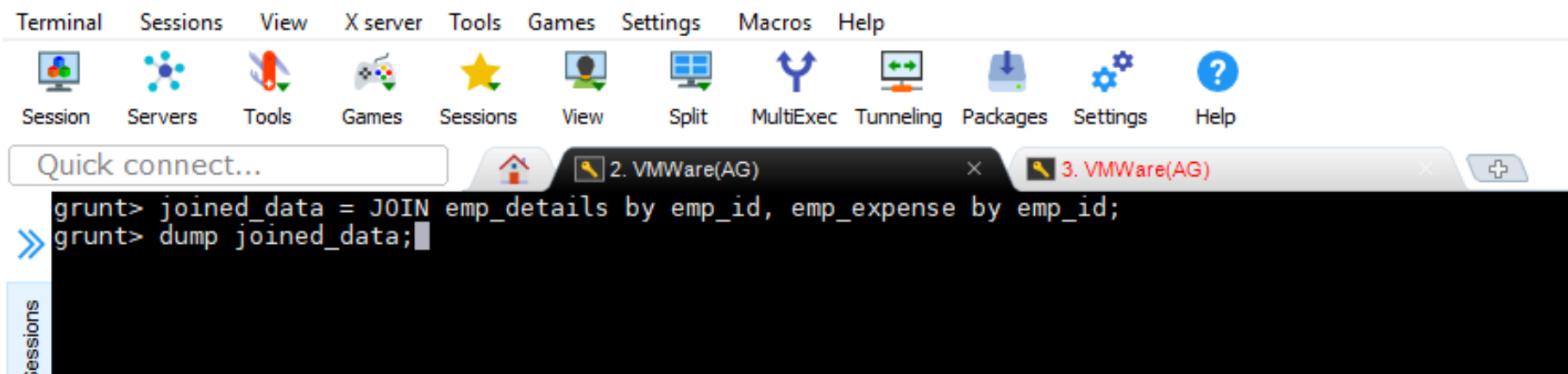
Load the emp_expense.txt file using PigStorage



Contents of file-



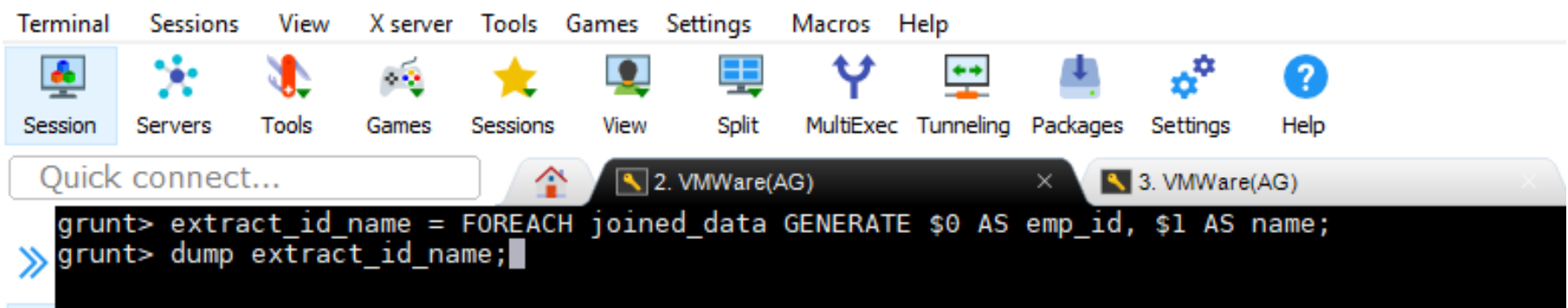
Join both files using JOIN relation-



Result after JOIN

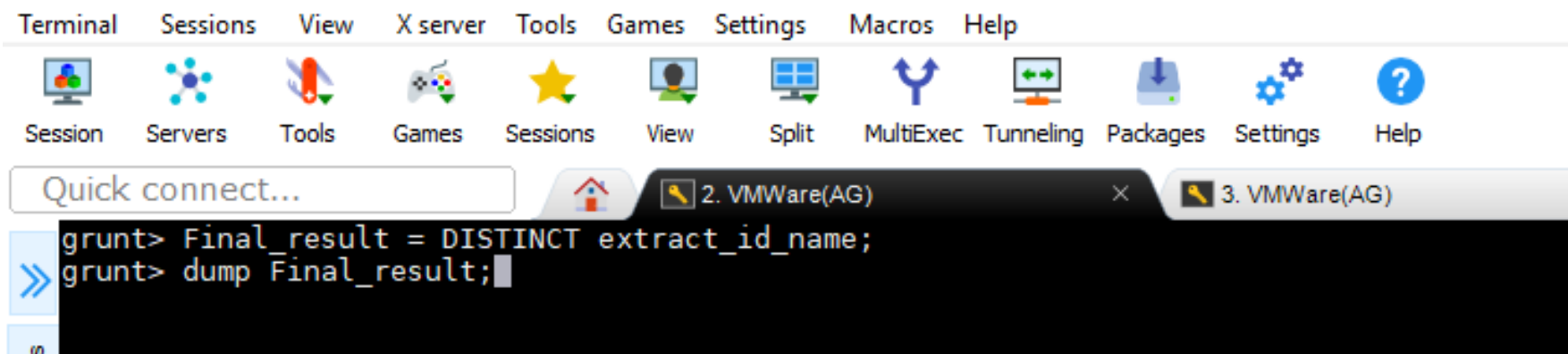
```
2017-10-23 22:45:06,477 [main] WARN org.apache.pig.data.SchemaTupleBackend - SchemaTupleBackend has already been initialized
2017-10-23 22:45:06,541 [main] INFO org.apache.hadoop.mapreduce.lib.input.FileInputFormat - Total input paths to process : 1
2017-10-23 22:45:06,542 [main] INFO org.apache.pig.backend.hadoop.executionengine.util.MapRedUtil - Total input paths to process : 1
(101,Amitabh,20000,1,101,100)
(101,Amitabh,20000,1,101,200)
(102,Shahrukh,10000,2,102,400)
(102,Shahrukh,10000,2,102,100)
(104,Anubhav,5000,4,104,300)
(105,Pawan,2500,5,105,100)
(110,Priyanka,2000,5,110,400)
(114,Madhuri,2000,2,114,200)
grunt>
```

Extract EMP_ID and NAME as required-



```
2017-10-24 16:05:24,992 [main] INFO org.apache.pig.backend.hadoop.executionengine.util.MapRedUtil - Total input paths to process : 1
(101,Amitabh)
(101,Amitabh)
(102,Shahrukh)
(102,Shahrukh)
(104,Anubhav)
(105,Pawan)
(110,Priyanka)
(114,Madhuri)
grunt>
```

Take DISTINCT EMP_IDs -



FINAL O/P-

```
2017-10-24 16:06:33,764 [main] WARN org.apache.pig.data.SchemaTupleBackend - SchemaTupleBackend has already been initialized
2017-10-24 16:06:33,798 [main] INFO org.apache.hadoop.mapreduce.lib.input.FileInputFormat - Total input paths to process : 1
2017-10-24 16:06:33,798 [main] INFO org.apache.pig.backend.hadoop.executionengine.util.MapRedUtil - Total input paths to process : 1
(101,Amitabh)
(102,Shahrukh)
(104,Anubhav)
(105,Pawan)
(110,Priyanka)
(114,Madhuri)
grunt>
```

e. List of employees (employee id and employee name) having no entry in employee expenses file.

Below is the script used to finalize the result-

Pig -x local;

emp_details = LOAD '/home/acadgild/pig/employee_details.txt' USING PigStorage(',') AS (emp_id:int, name:chararray, salary:int, rating:int);

emp_expense = LOAD '/home/acadgild/pig/employee_expenses.txt' USING PigStorage('\t') AS (emp_id:int, expense:int);

joined_data = JOIN emp_details by emp_id FULL, emp_expense by emp_id;

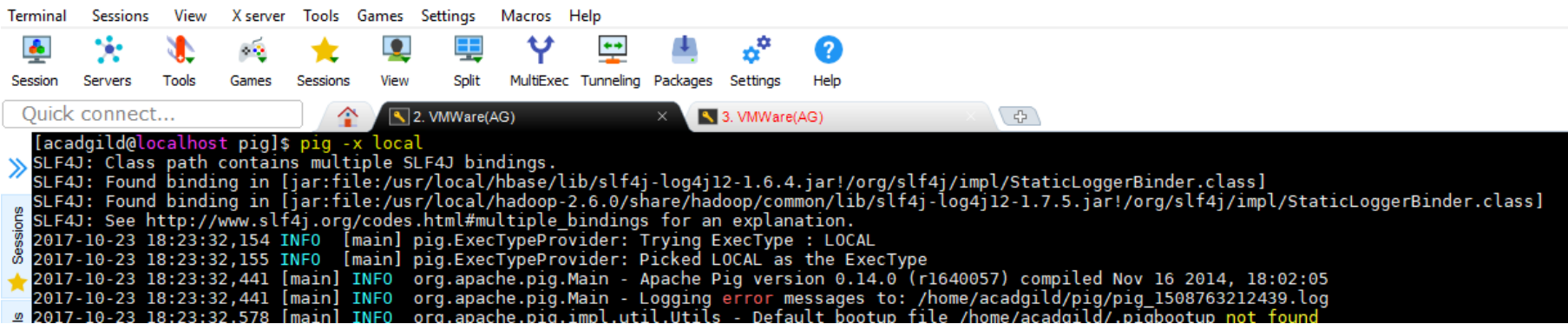
filter_res = FILTER joined_data BY emp_expense::emp_id IS NULL;

final_result = FOREACH filter_res GENERATE \$0, \$1;

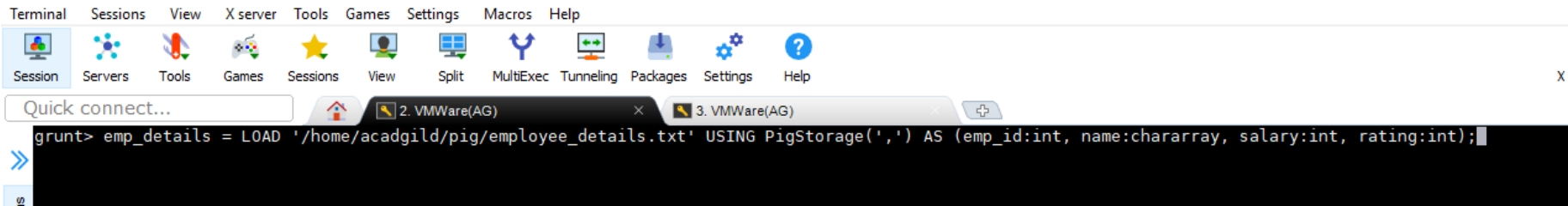
dump final_result;

Each and every relation has been explained below with its immediate corresponding output-

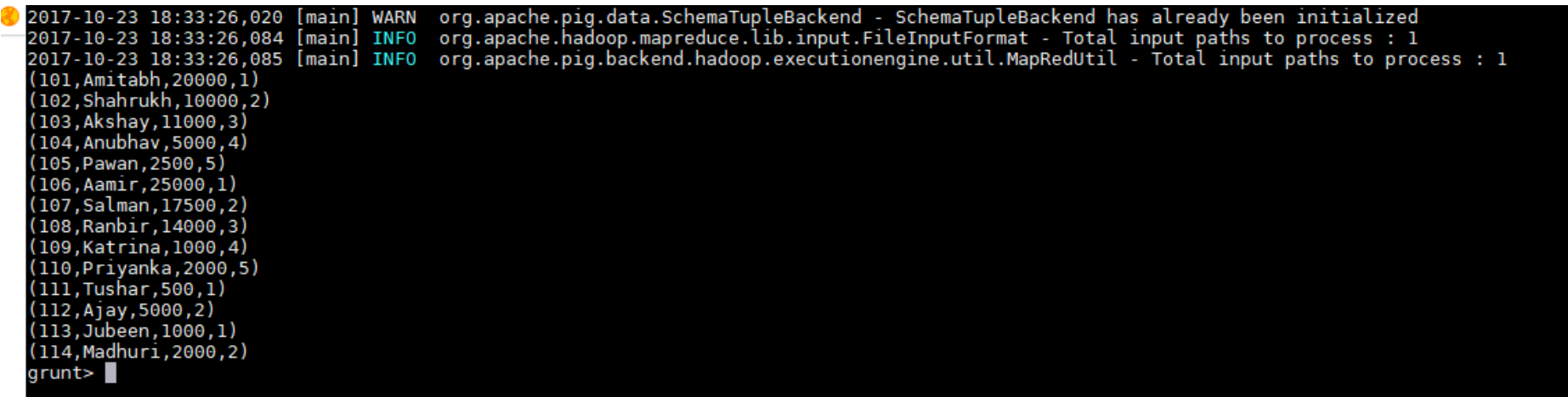
Start pig in local mode-



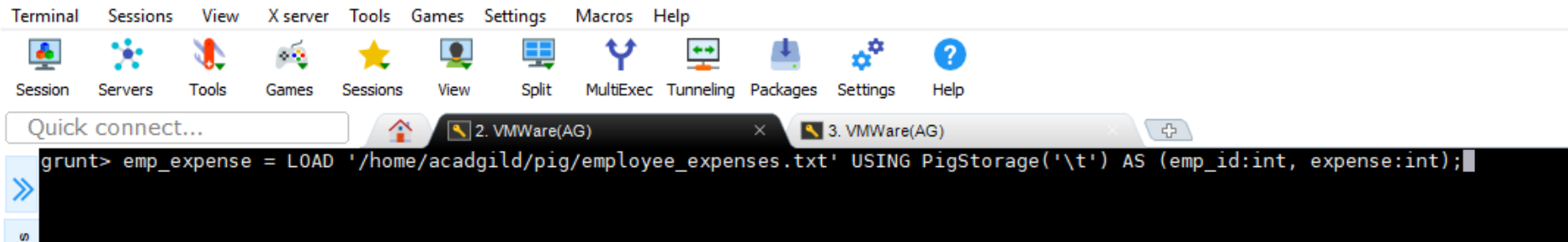
Load the emp_details.txt file using PigStorage



Contents of file-



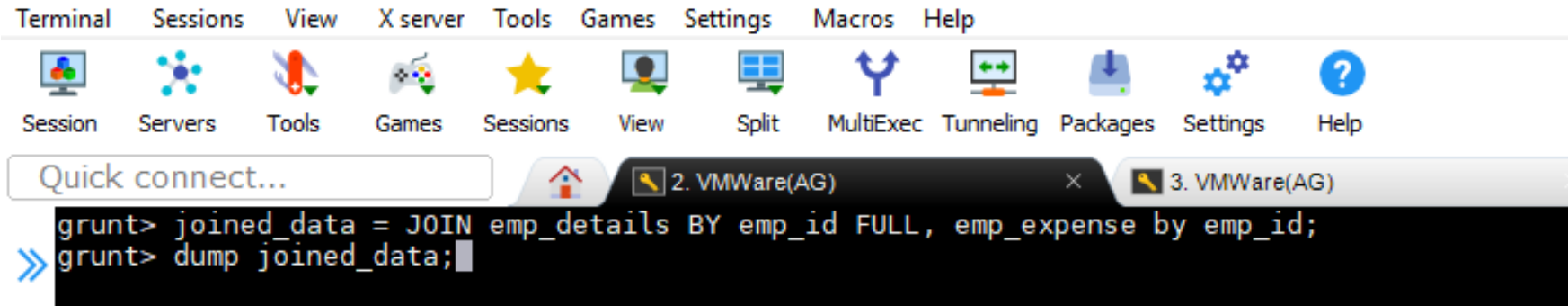
Load the emp_expense.txt file using PigStorage



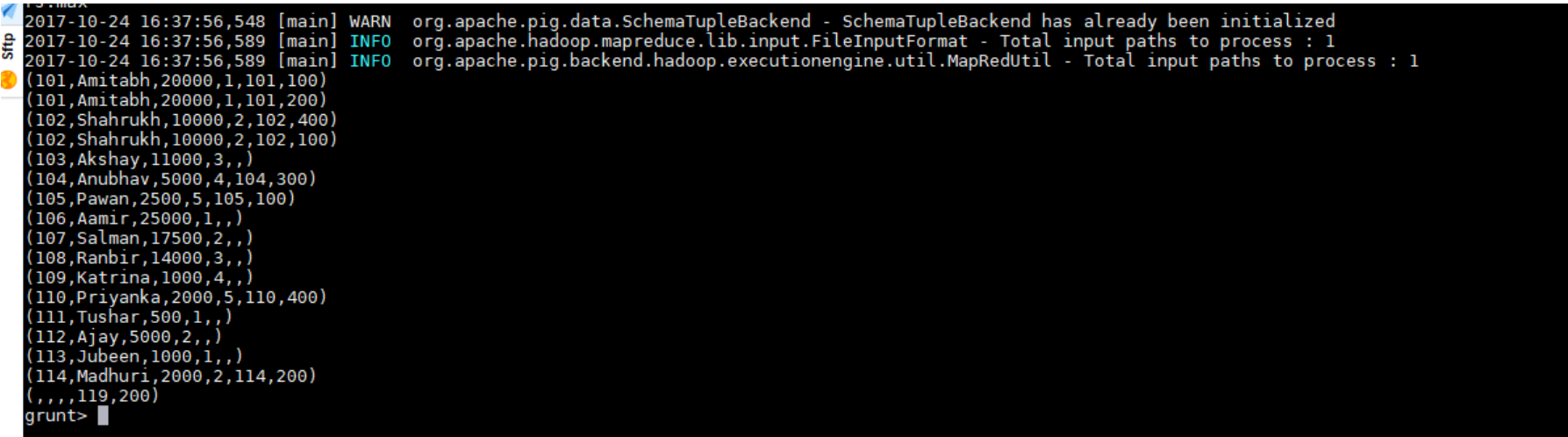
Contents of file-

```
2017-10-23 22:33:51,736 [main] WARN org.apache.pig.data.SchemaTupleBackend - SchemaTupleBackend has already been initialized
2017-10-23 22:33:51,808 [main] INFO org.apache.hadoop.mapreduce.lib.input.FileInputFormat - Total input paths to process : 1
2017-10-23 22:33:51,811 [main] INFO org.apache.pig.backend.hadoop.executionengine.util.MapRedUtil - Total input paths to process : 1
(101,200)
(102,100)
(110,400)
(114,200)
(119,200)
(105,100)
(101,100)
(104,300)
(102,400)
grunt>
```

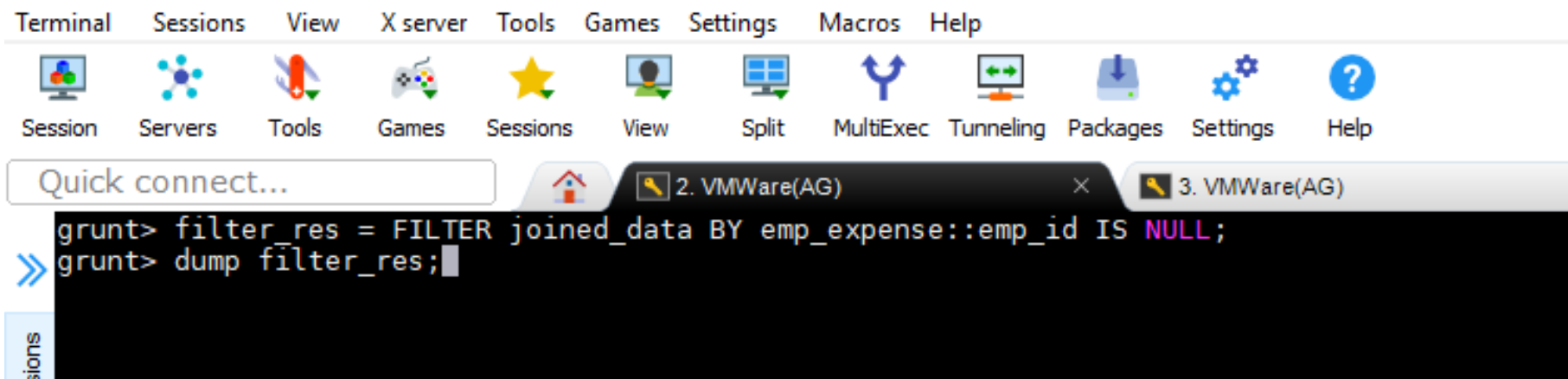
Do a FULL OUTER join on both files -



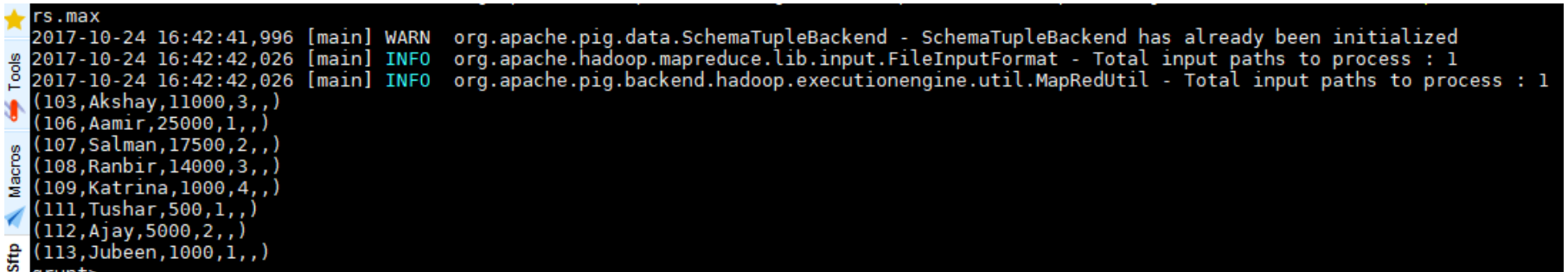
Result of FULL OUTER JOIN-



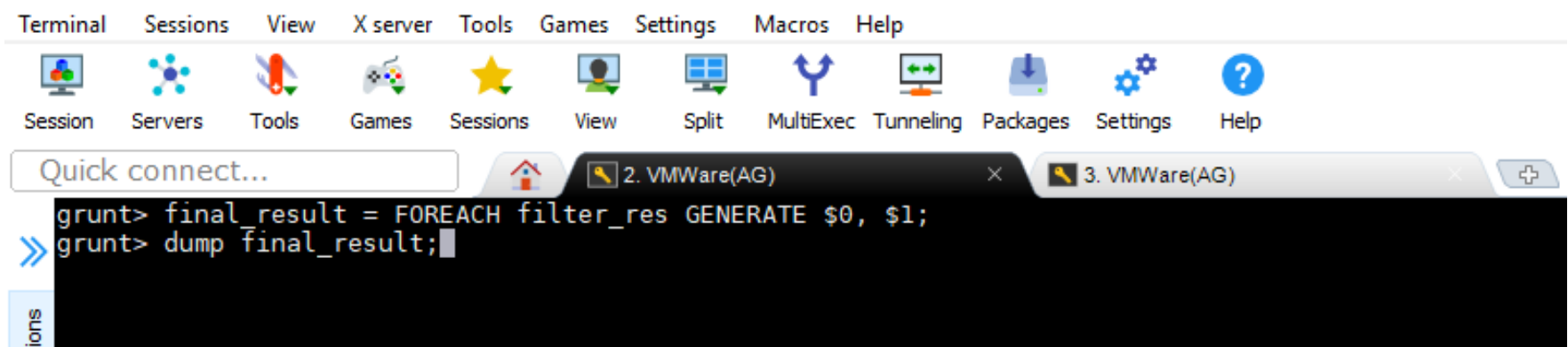
Check whether contents(emp_id) from second file(emp_expense) is NULL and FILTER them-



Result of Filtered out data which are not present in emp_expense file-



Extract emp_id aand emp_name from first file as required



Final O/P-

