Working of pigscript commands :

1. LOAD

It is a relational operator used To Load the data from the file system (local/HDFS) into a relation.

SYNTAX:

LOAD 'data' [USING function] [AS schema];

WHERE

Data - The name of the file or directory, in single quotes.

If you specify a directory name, all the files in the directory are loaded.

USING – keyword

If the USING clause is omitted, the default load function PigStorage is used.

Function - The load function.

* You can use a built-in function (Load/Store Functions). PigStorage is the default load function and does not need to be specified (simply omit the USING clause).
* You can write your own load function if your data is in a format that cannot be processed by the built-in functions

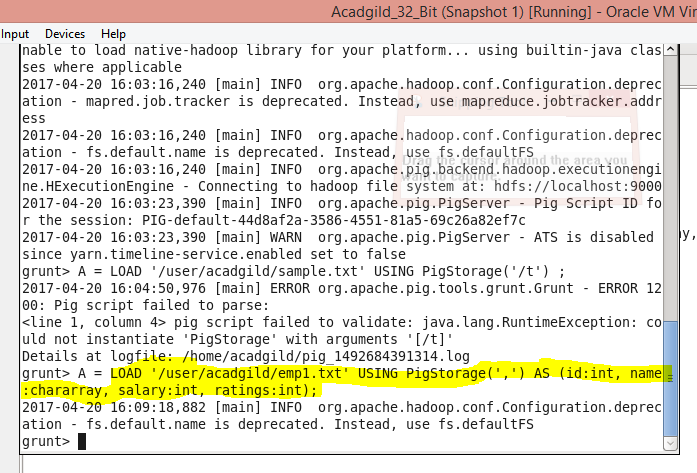
AS –Keyword

Schema -A schema using the AS keyword, enclosed in parentheses (see Schemas).

The loader produces the data of the type specified by the schema. If the data does not conform to the schema, depending on the loader, either a null value or an error is generated.

Example :

A = LOAD ‘/user/acadgild/emp1.txt’ USING PigStorage (‘,’) AS (id:int,name:chararray,salary:int,ratings:int) ;



Screenshot of terminal where load command is executed

2.STORE

It is a relational operator used To save a relation to the file system (local/HDFS).

SYNTAX :

STORE alias INTO 'directory' [USING function];

WHERE

alias - The name of a relation.

INTO - Required keyword.

Directory - The name of the storage directory, in quotes. If the directory already exists, the STORE operation will fail.

The output data files, named part-nnnnn, are written to this directory.

USING - Keyword. Use this clause to name the store function.

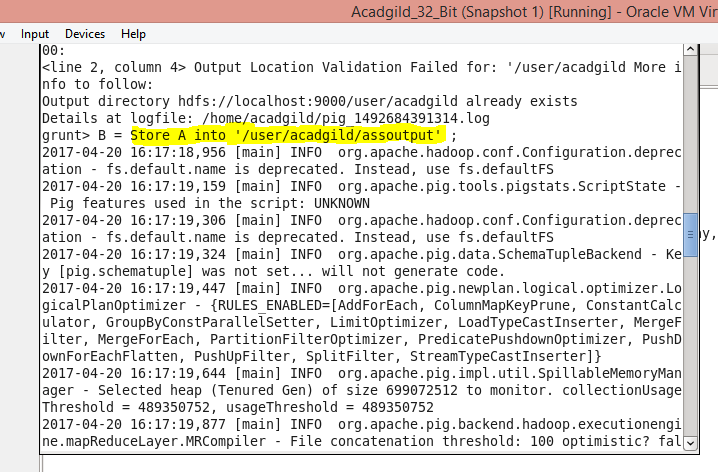
If the USING clause is omitted, the default store function PigStorage is used.

Function - The store function.

* You can use a built-in function (Load/Store Functions). PigStorage is the default store function and does not need to be specified .
* You can write your own store function if your data is in a format that cannot be processed by the built-in functions

EXAMPLE :

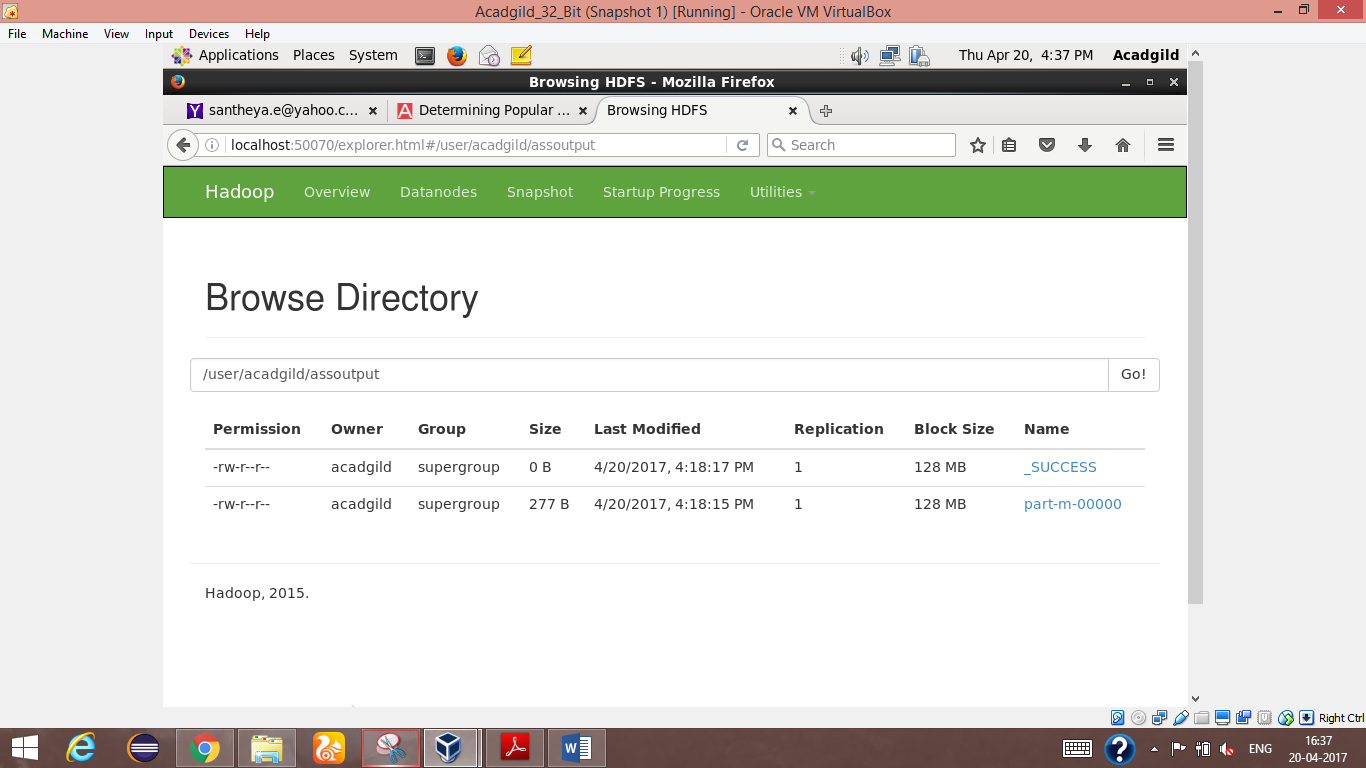
B = STORE A into ‘/user/acadgild/assoutput’ ;



Screen shot of the terminal where store is running



Terminal where store run successfully



Destination where it is stored

3.DUMP

It is used To print the contents of a relation on the console .

SYNTAX

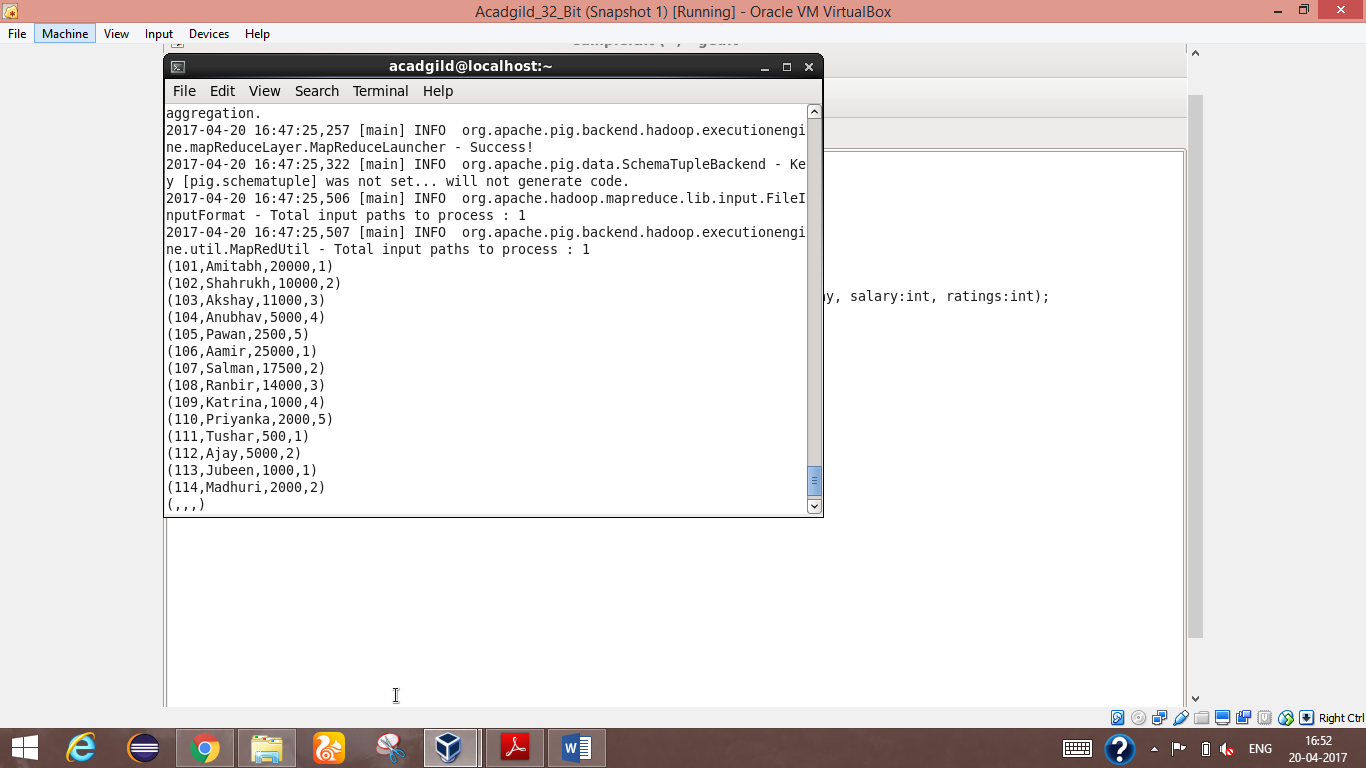
DUMP alias ;

WHERE

Alias - The name of a relation.

Example :

DUMP A ;



Screenshot where dump result is displayed

4.FOREACH

It is used To generate data transformations based on columns of data.

SYNTAX :

alias  = FOREACH { gen\_blk | nested\_gen\_blk } [AS schema];

WHERE

Alias - The name of relation

gen\_blk - FOREACH … GENERATE used with a relation (outer bag). Use this syntax:

alias = FOREACH alias GENERATE expression [expression ….]

nested\_gen\_blk - FOREACH … GENERATE used with a inner bag. Use this syntax:

alias = FOREACH nested\_alias {

   alias = nested\_op; [alias = nested\_op; …]

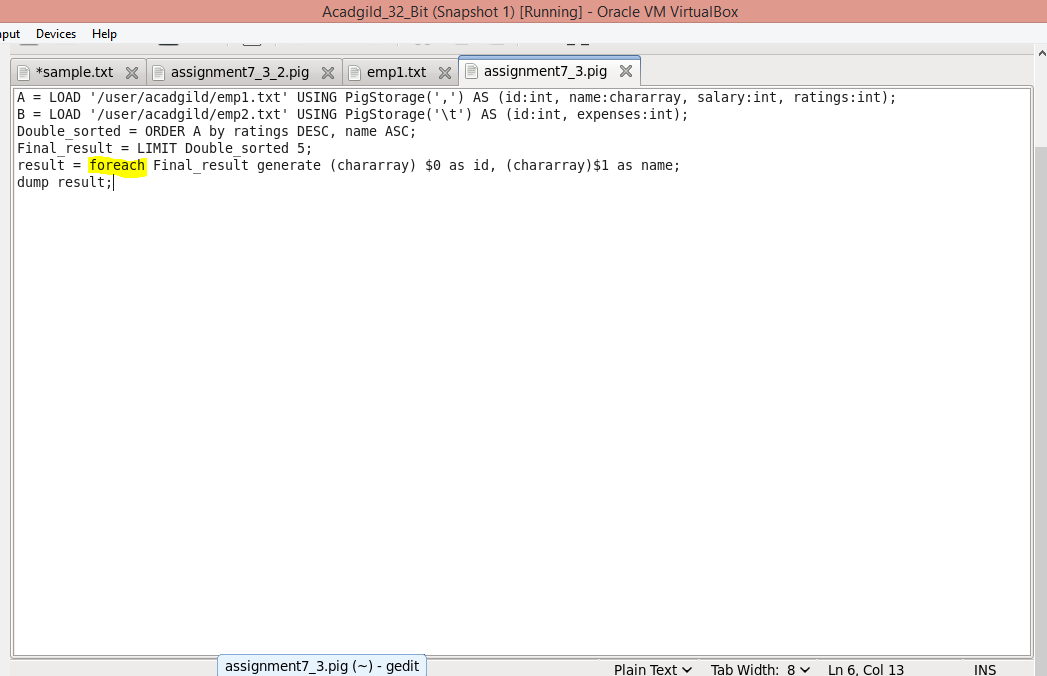
   GENERATE expression [, expression …]

};

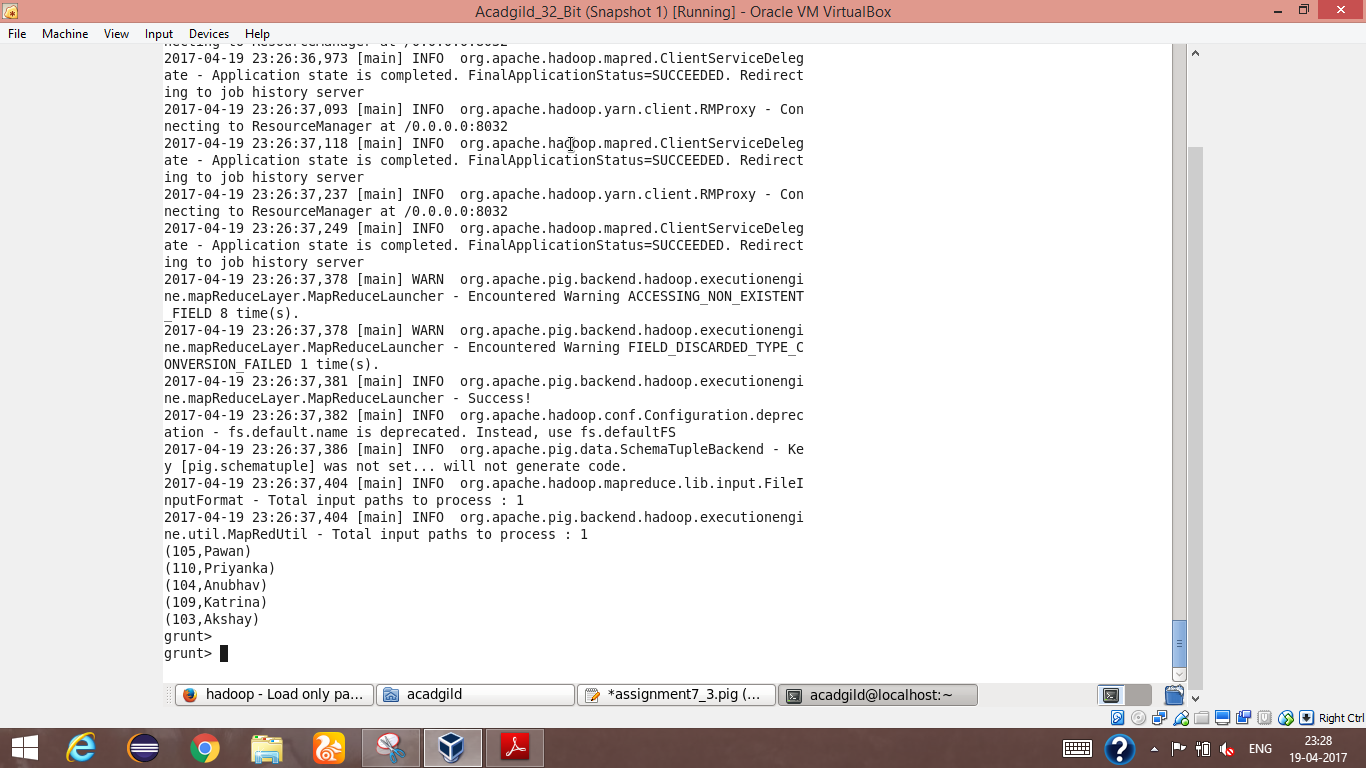
Where:

The nested block is enclosed in opening and closing brackets { … }.

The GENERATE keyword must be the last statement within the nested block.



Screen where foreach command is used



Screen where the result of the above shown script

5.FILTER

It is used to To remove unwanted rows from a relation.

SYNTAX :

alias = FILTER alias  BY expression;

WHERE

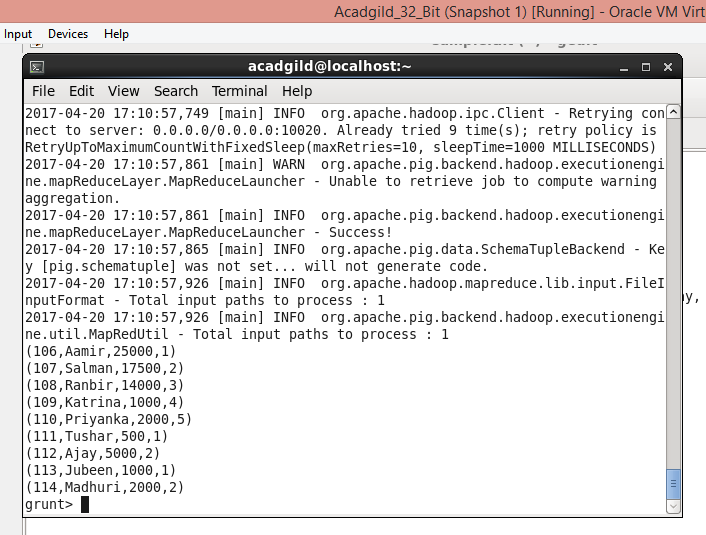
Alias - The name of the relation.

BY- Required keyword

Expression - A boolean expression.

Example :

D = filter A by id >105 ;

 Screen where the output of the filter command result

6.GROUP BY

It is used to To group the data in a single relation .

SYNTAX :

alias = GROUP alias { ALL | BY expression}

WHERE

Alias – Name of the relation

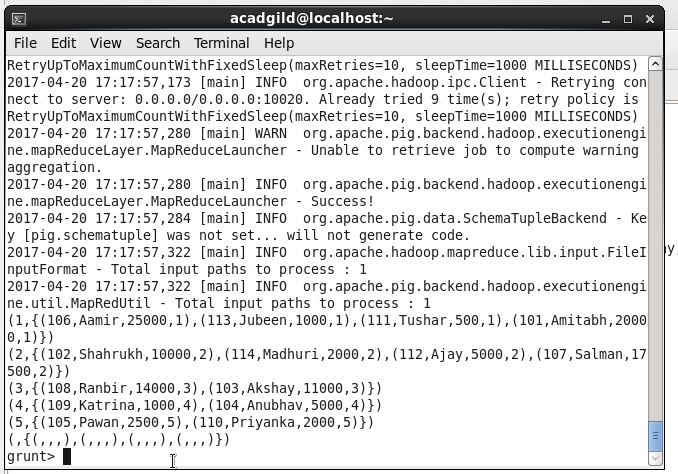
ALL - Keyword. Use ALL if you want all tuples to go to a single group; for example, when doing aggregates across entire relations.

BY - Keyword. Use this clause to group the relation by field, tuple or expression

Expression - A tuple expression. This is the group key or key field. If the result of the tuple expression is a single field, the key will be the value of the first field rather than a tuple with one field. To group using multiple keys, enclose the keys in parentheses.

EXAMPLE :

B = group A by ratings ;



Screen where the output of group command is shown

7.ORDER BY

It is used to To arrange a relation in a sorted order based on one or more fields (ascending or descending).

SYNTAX :

alias = ORDER alias BY [ASC|DESC]

WHERE

Alias – Name of the relation

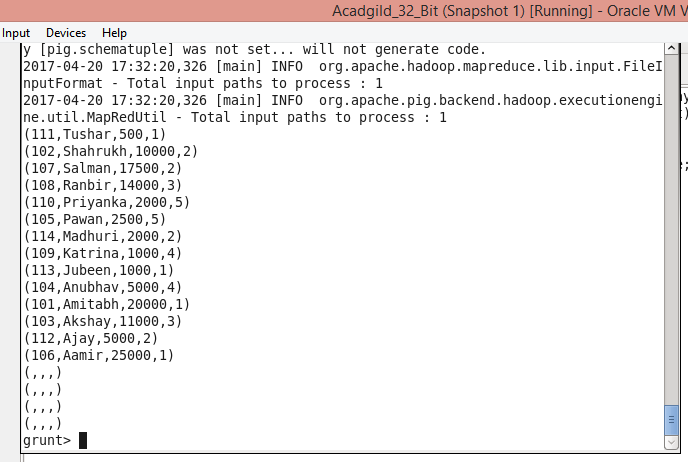
BY – required keyword

ASC – In ascending order

DESC – In descending order

EXAMPLE :

C = order A by name DESC ;



Screenshot where the above command result is shown

8.DESCRIBE

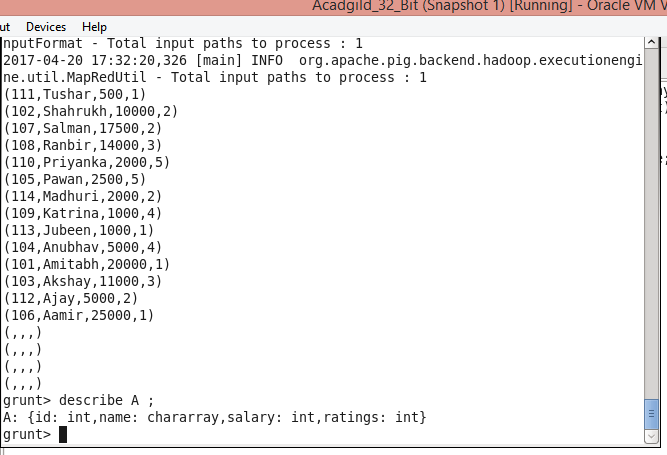
It is used to To describe the schema of a relation.

SYNTAX :

DESCRIBE alias ;

EXAMPLE :

Describe C ;



Screenshot were output of describe is used