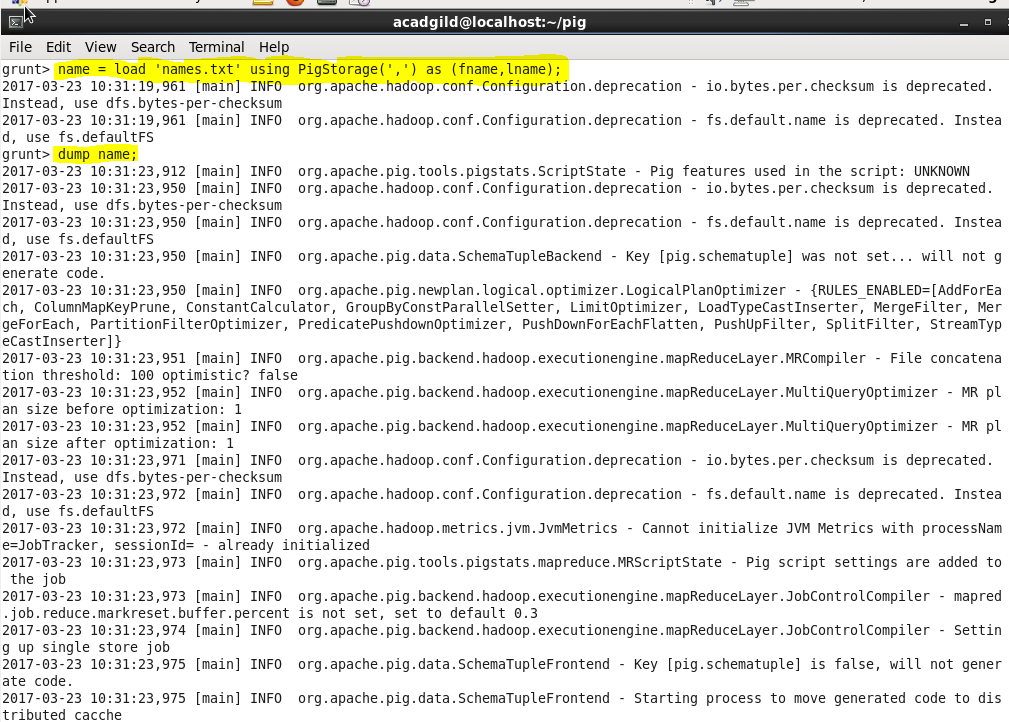
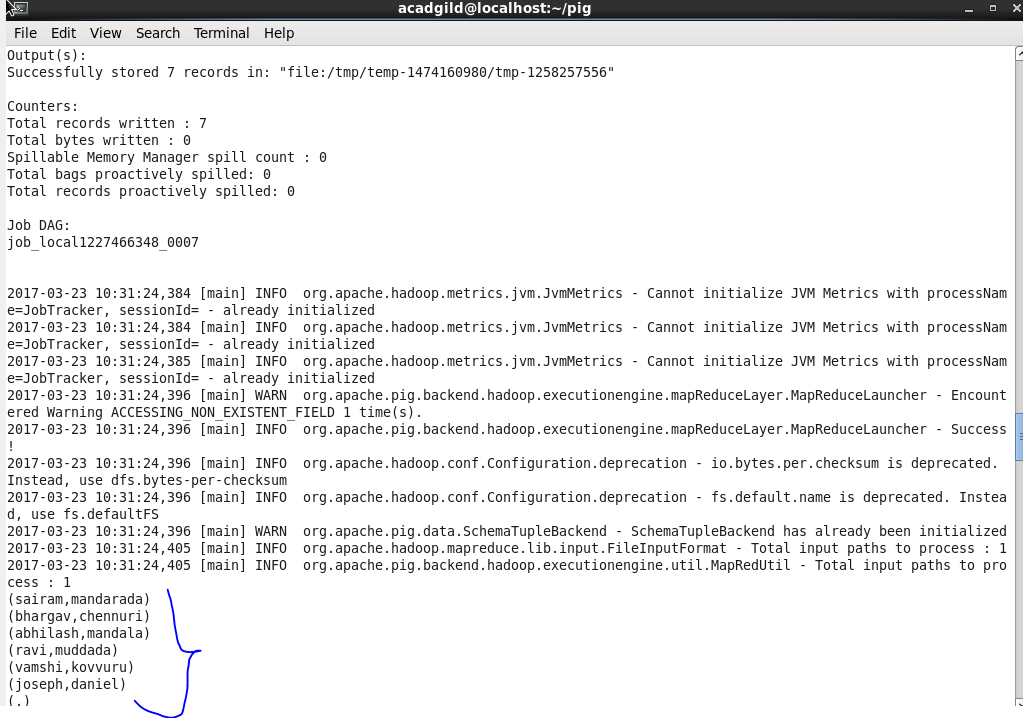
**Assignment 10.2**

**Question:** Create a sample dataset and implement the below Pig commands on the same dataset.

* Concat
* Tokenize
* Sum
* Min
* Max
* Limit
* Store
* Distinct
* Flatten
* IsEmpty



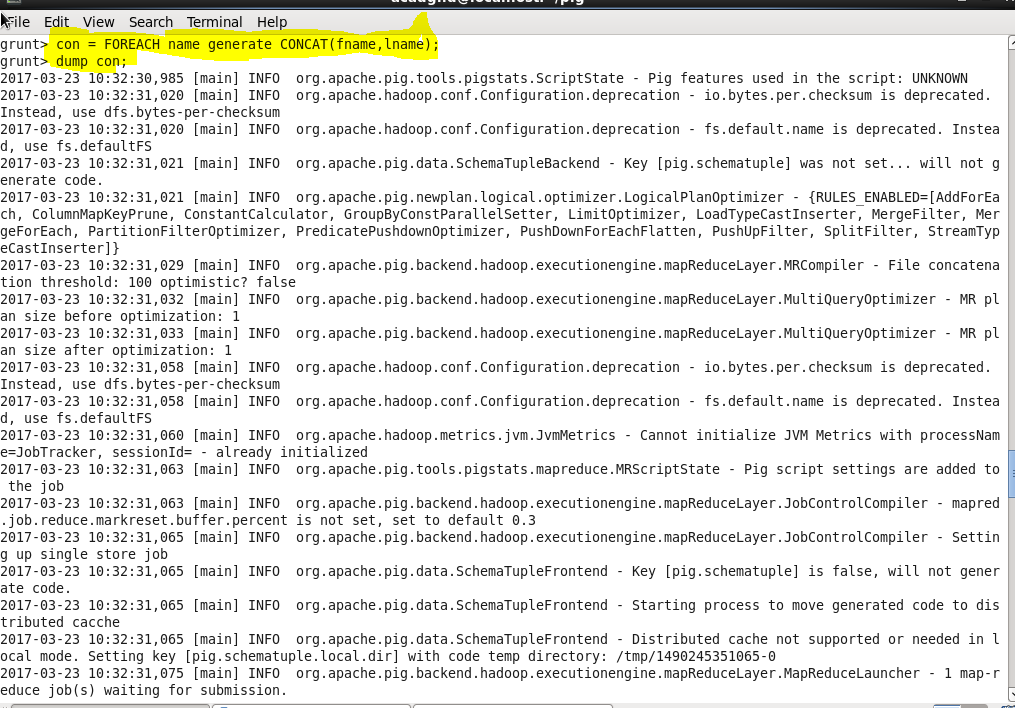


**Concat:**

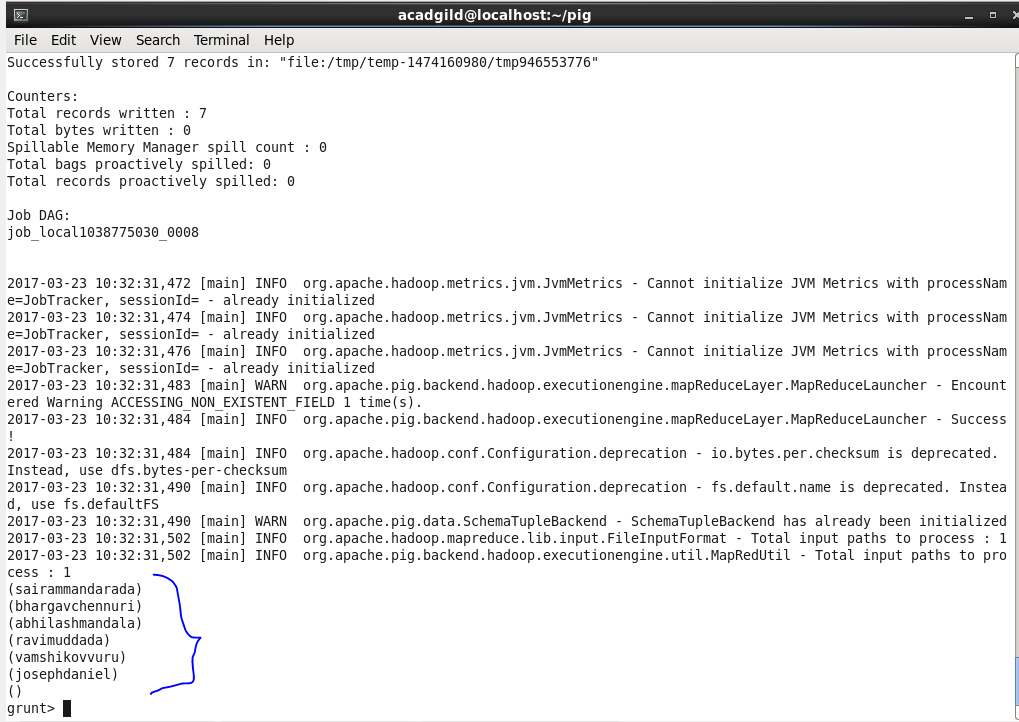
The *CONCAT()* function of Pig Latin is used to concatenate two or more expressions of the same type.

*Syntax:*

***CONCAT(expression,expression,[expression]);***



**OUTPUT:**

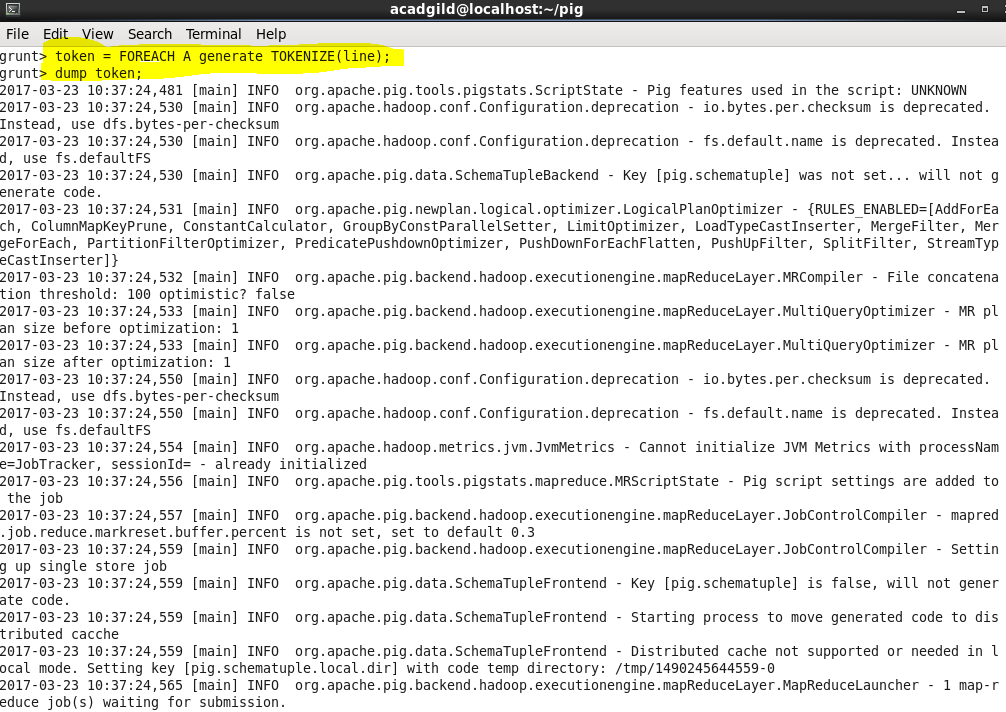


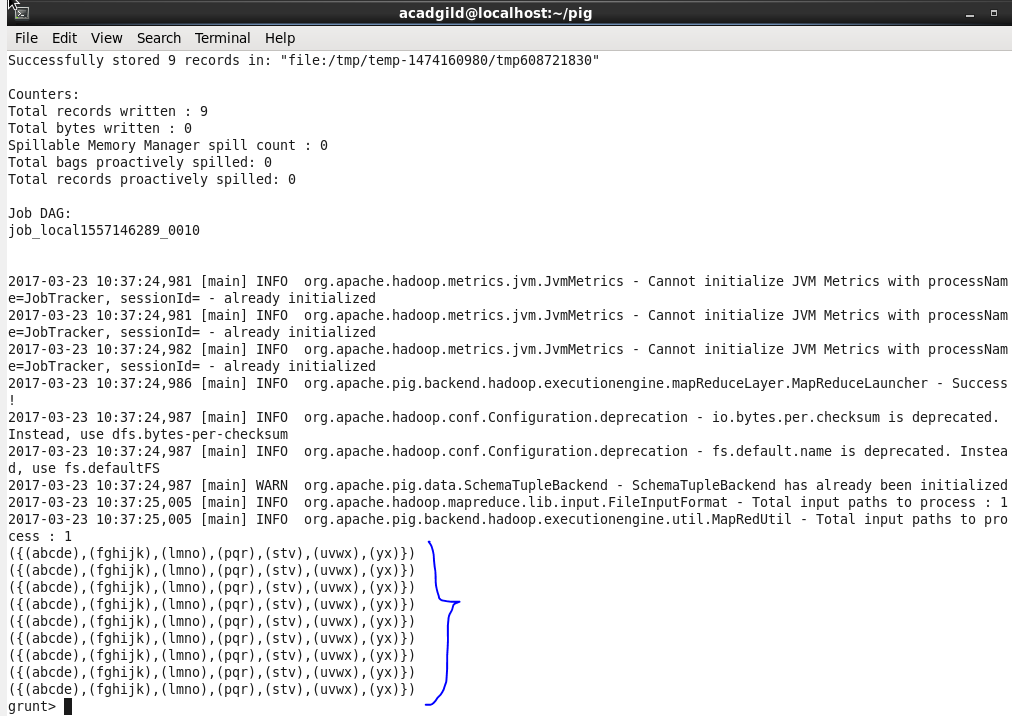
**Tokenize:**

The *TOKENIZE***()** function of Pig Latin is used to split a string (which contains a group of words) in a single tuple and returns a bag which contains the output of the split operation.

*Syntax:*

***TOKENIZE(expression,[field\_delimiter’]);***

******

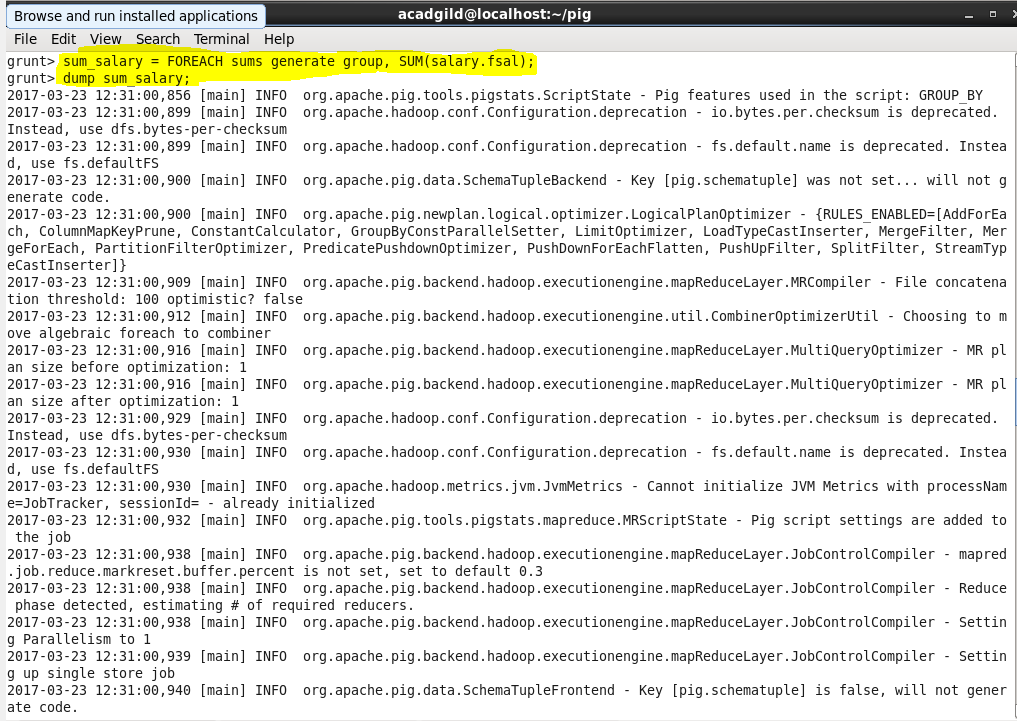
******

**Sum*:***

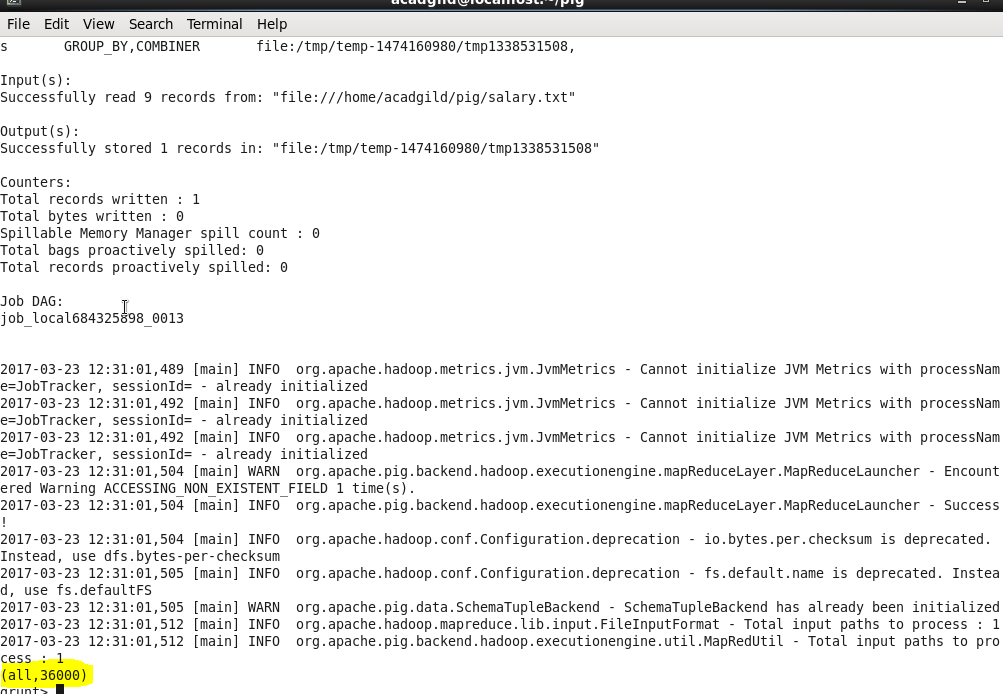
We can use the *SUM()* function of Pig Latin to get the total of the numeric values of a column in a single-column bag. While computing the total, the *SUM()* function ignores the NULL values.

*Syntax***:**

**SUM(expression);**

******

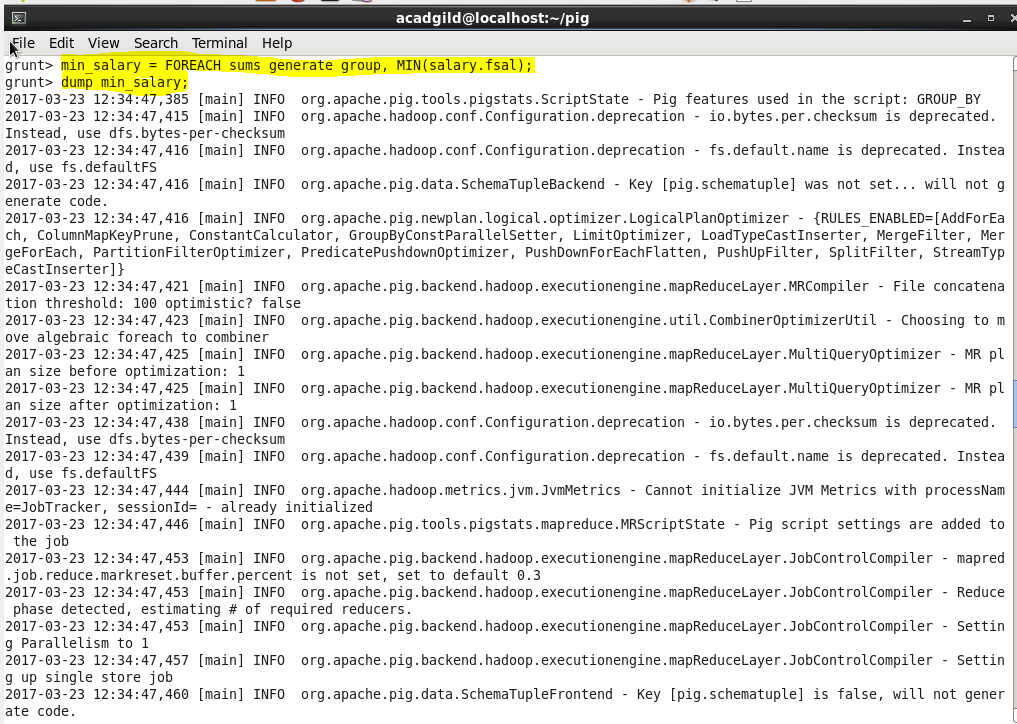
**OUTPUT:**

******

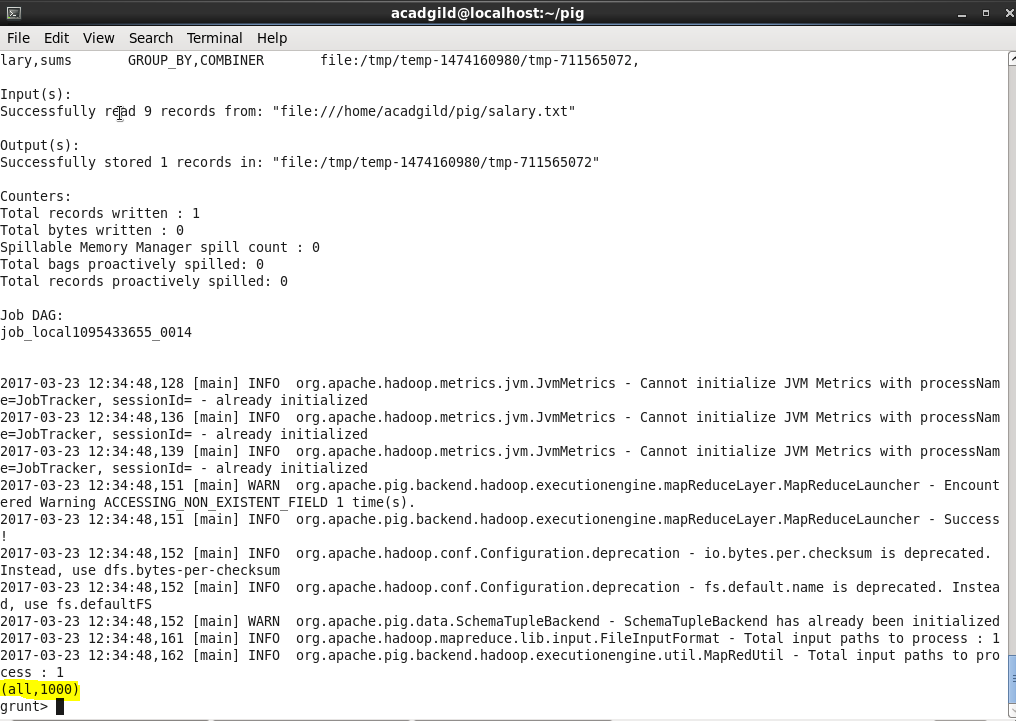
**Min:**

The *MIN()* function of Pig Latin is used to get the minimum (lowest) value (numeric or chararray) for a certain column in a single-column bag. While calculating the minimum value, the *MIN()* function ignores the NULL values.

*Syntax***:**

**MIN(expression);   
  
**

**OUTPUT:**

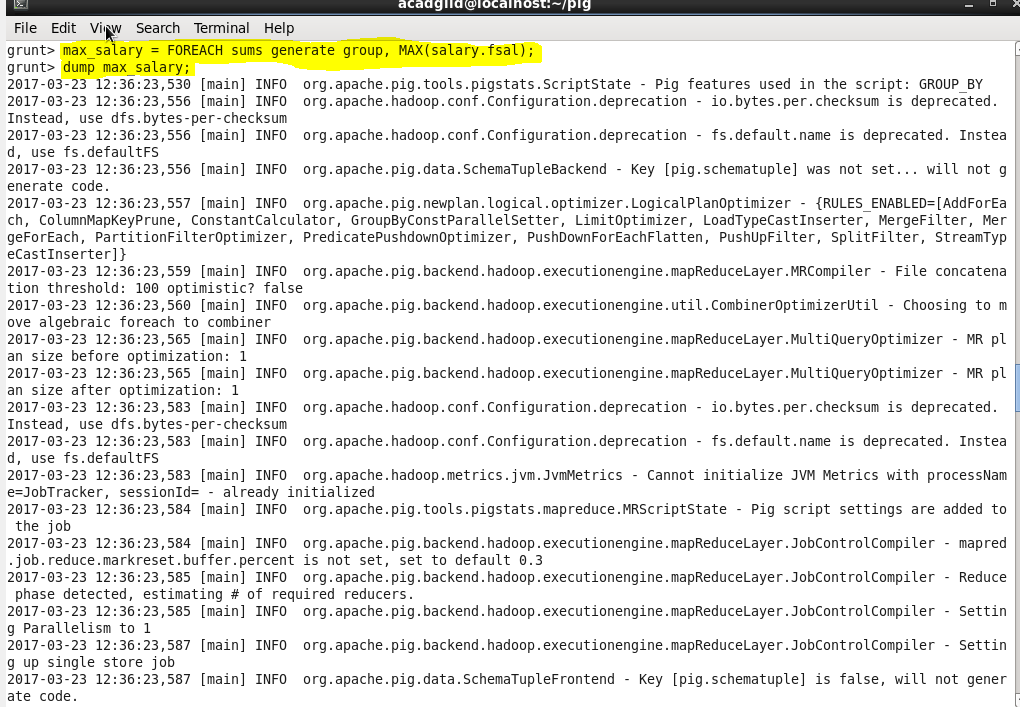
****

**Max:**

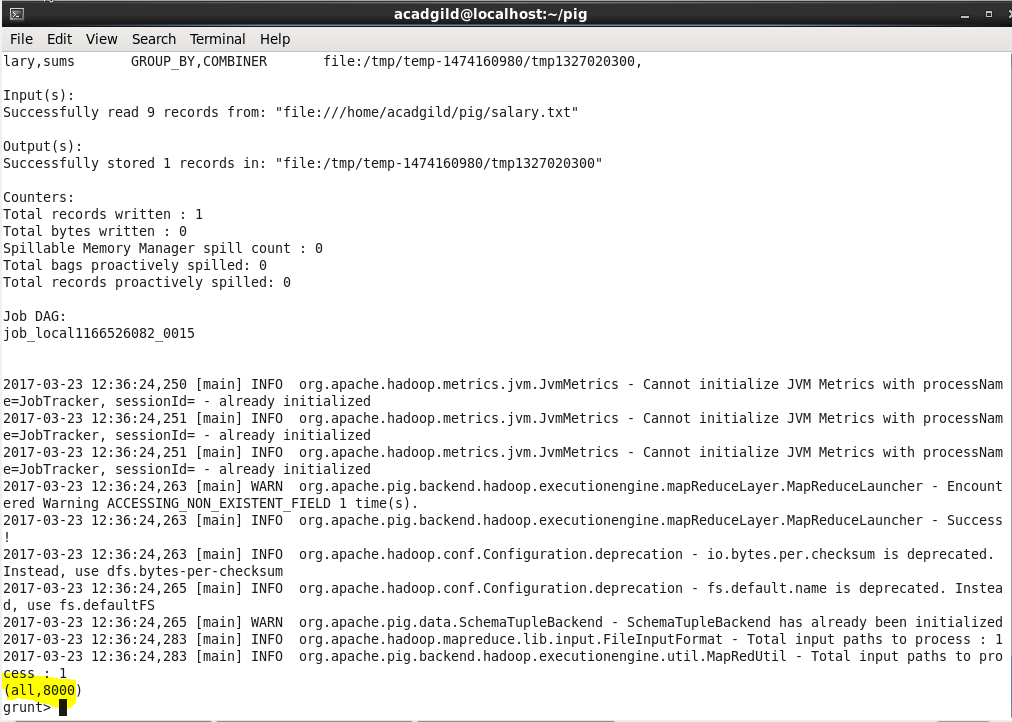
The Pig Latin *MAX()* function is used to calculate the highest value for a column (numeric values or chararrays) in a single-column bag. While calculating the maximum value, the *Max()* function ignores the NULL values.

***Syntax*:**

**MAX(expression);**

****

**OUTPUT:**

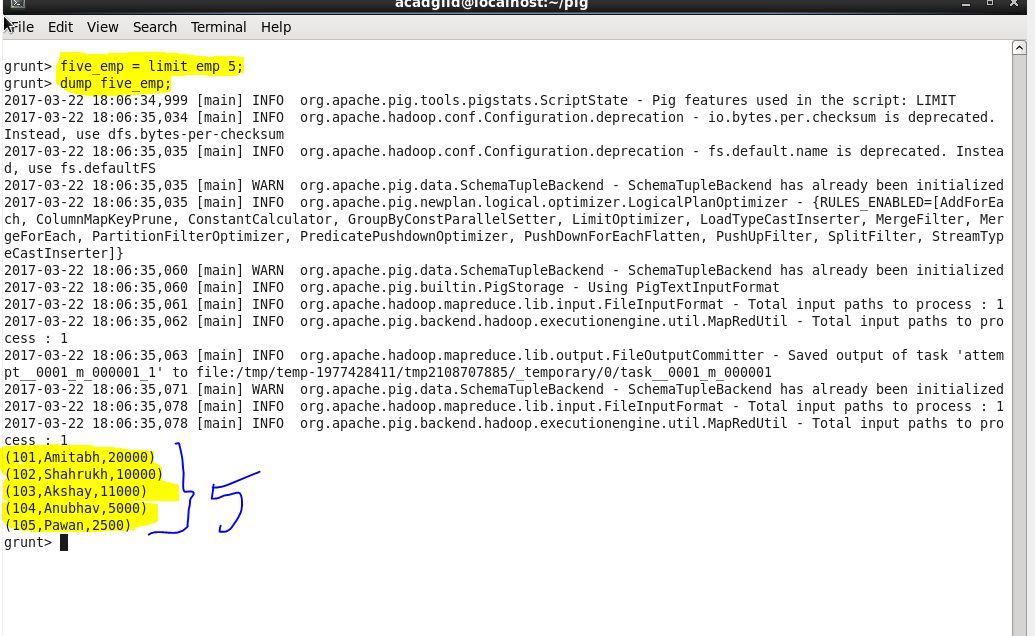
****

**Limit:**

Used to limit the number of outputs to the desired number.

*Syntax*:

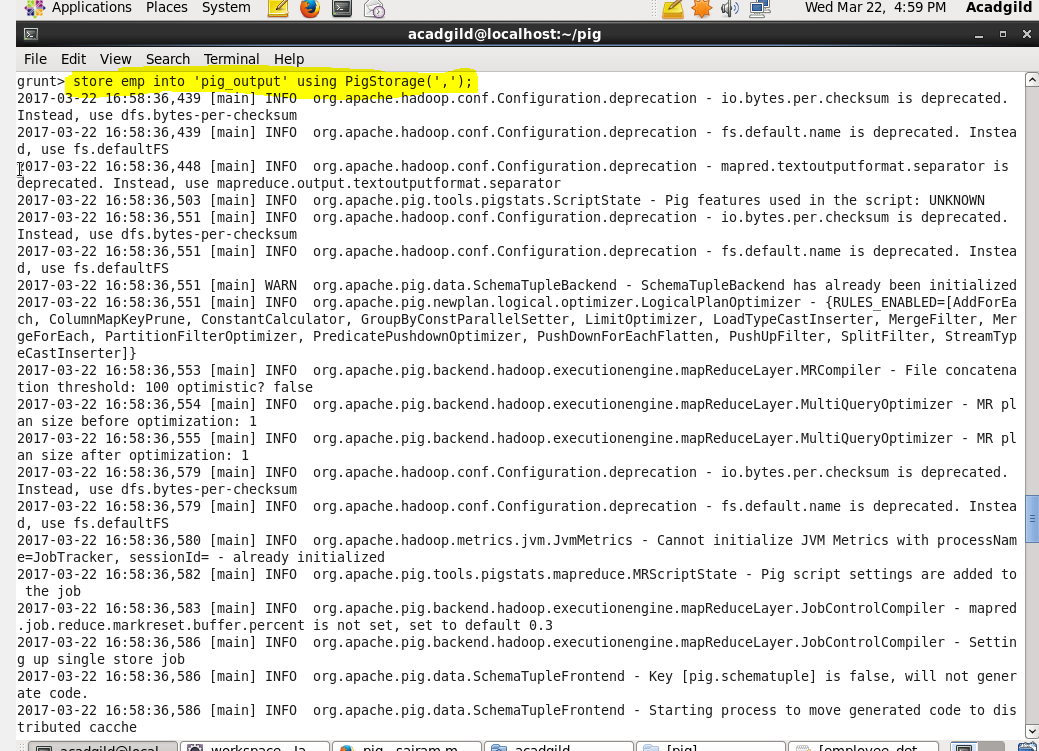
***Relation\_name2 = LIMIT Relation\_name1 n;***

****

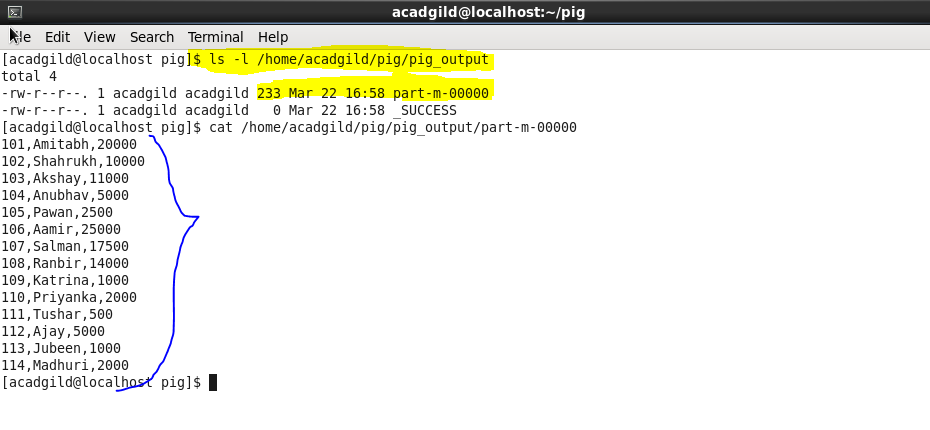
**STORE:**

We can store the loaded data in the file system using the *store* operator.  
*Syntax***:**

***STORE Relation\_name INTO ‘required\_directory\_path’ [USING Function];***

****

**OUTPUT:**

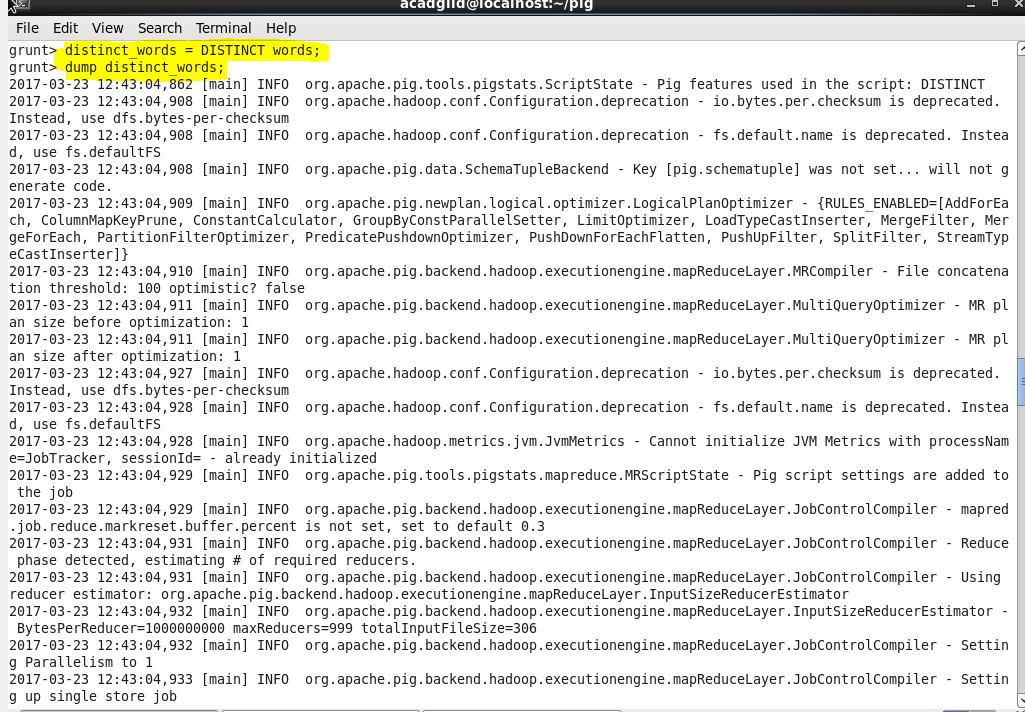
****

**Distinct:**

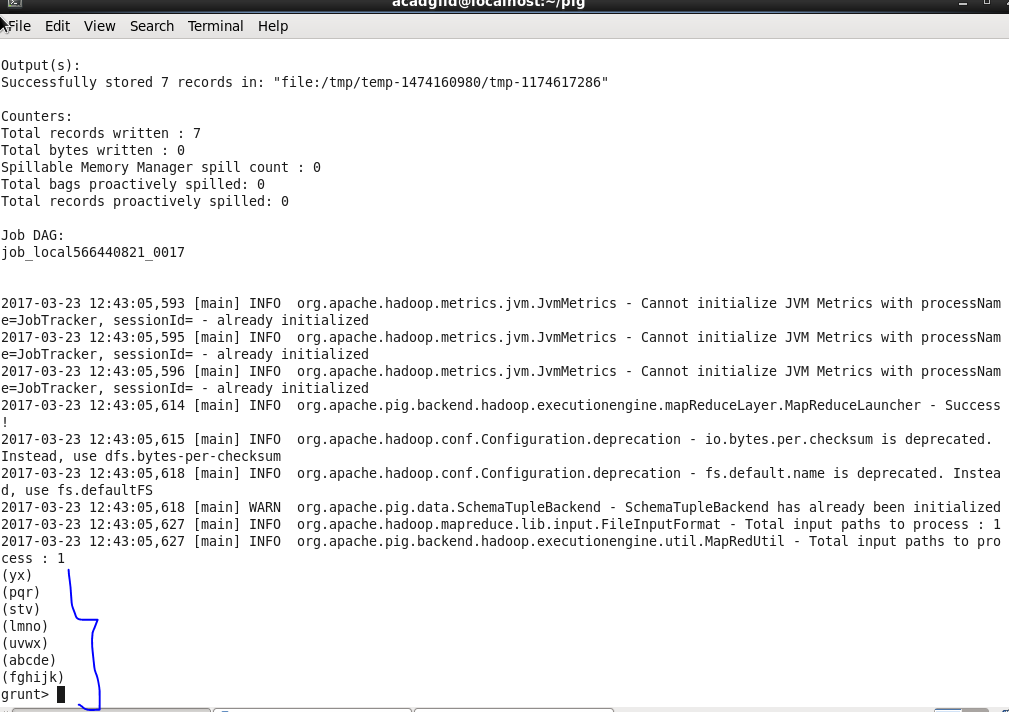
The *DISTINCT* operator is used to remove redundant (duplicate) tuples from a relation.

*Syntax:*

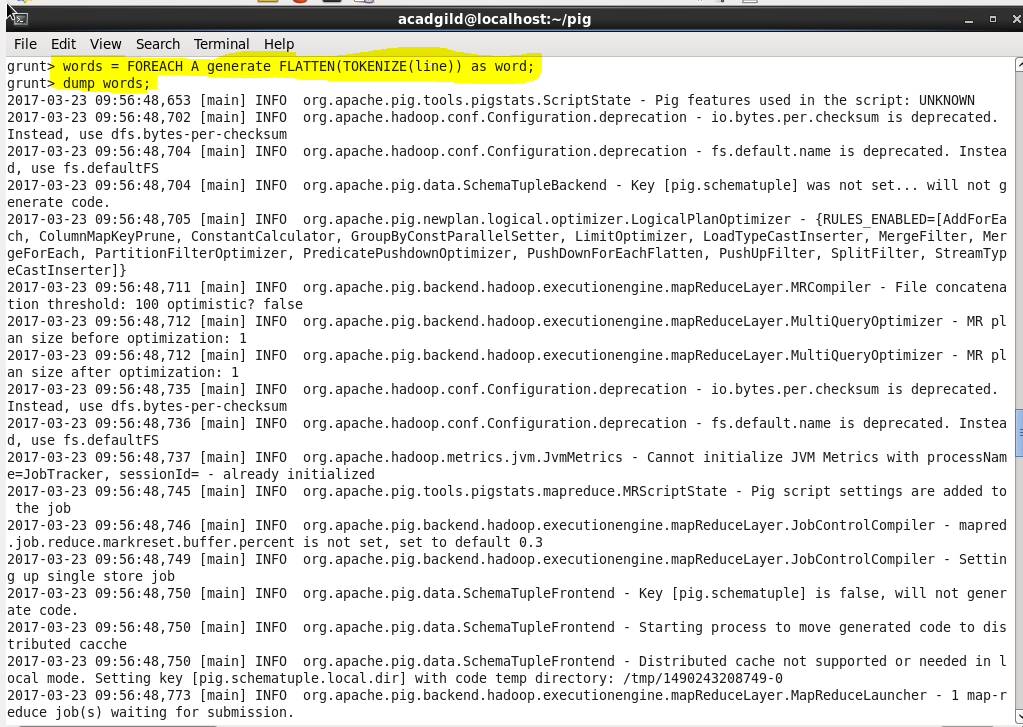
***Relation\_name2 = DISTINCT Relation\_name1;***

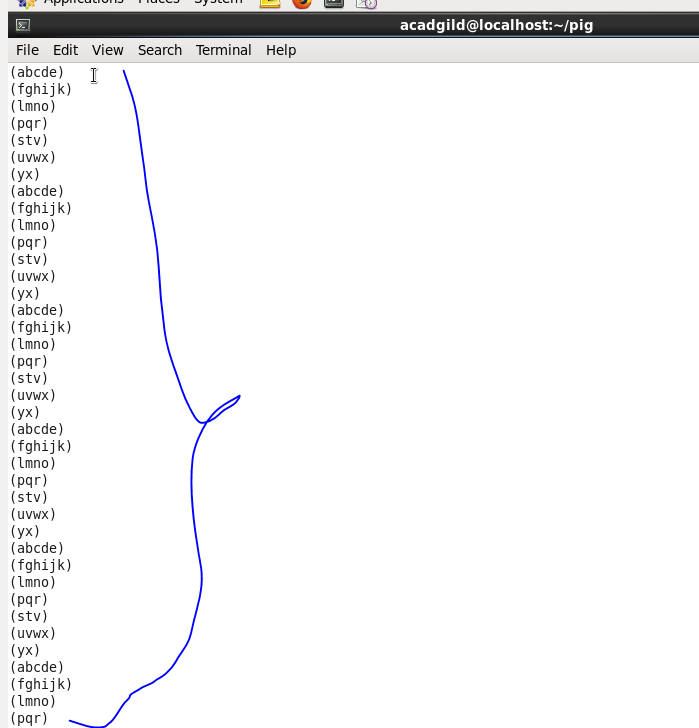
****

**OUTPUT:**

****

**Flatten:***FLATTEN* is an expression which will eliminate a level of nesting. Given a tuple which contains a bag, *FLATTEN* will emit several tuples each of which contains one record from the bag.

 **OUTPUT:**

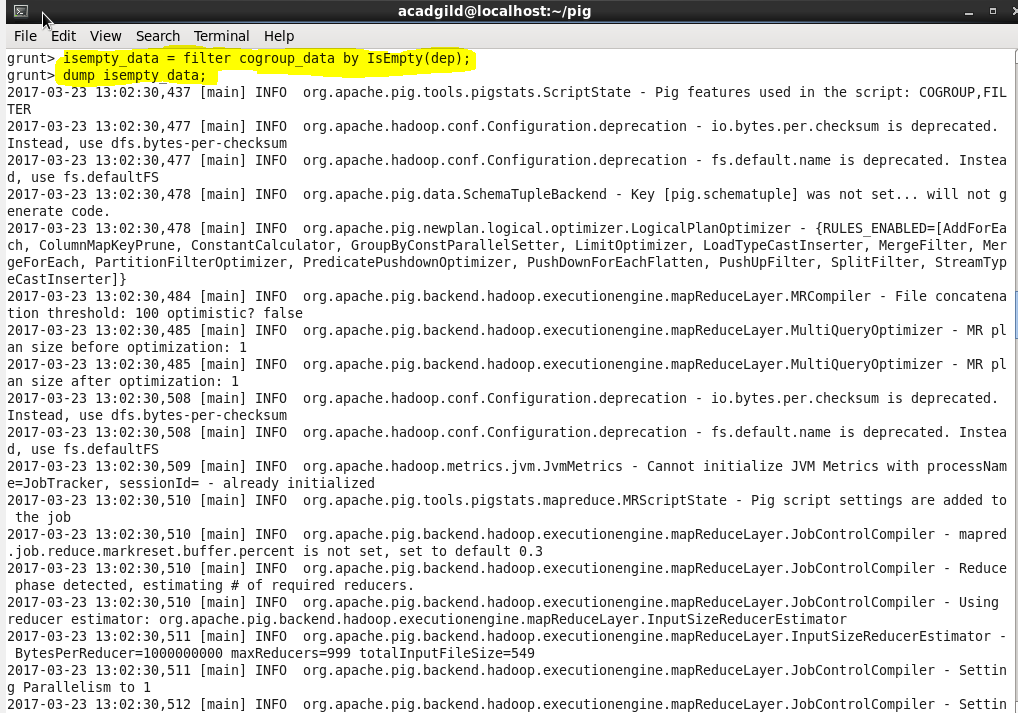


**IsEmpty:**

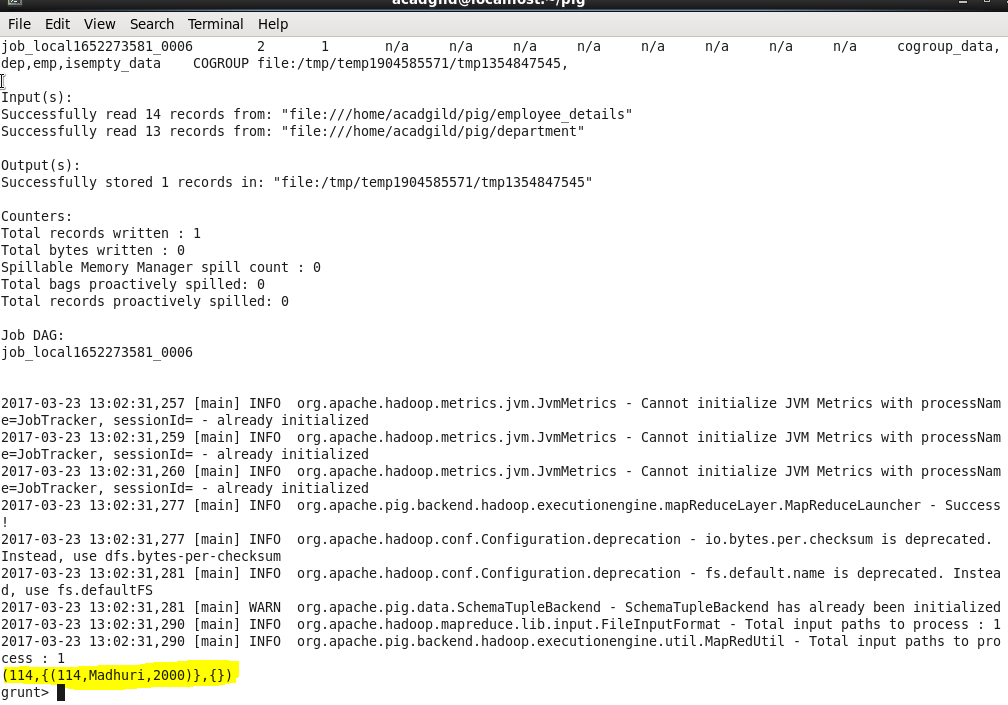
The *IsEmpty()* function of Pig Latin is used to check if a bag or map is empty.

*Syntax:*

***IsEmpty(expression);***

******

**Output:**

******