As pig lot of extra data, in this assignment, I have taken the only the final output of commands

Load datasets from students\_details.txt

Details of data set students\_details.txt is as below:

[acadgild@localhost assignment\_4.2]$ cat students\_details.txt

1,Ram Sharma,Delhi,22

2,Shyam Kumar,Mumbai,28

3,Madhu Patel,Ahmedabad,29

4,Rohit Jain,Bangalore,25

5,Biju Nair,Kochi,25

5,Biju Nair,Kochi,25

6,Hari Sharma,Kolkata,25

6,Hari Sharma,Kolkata,25

grunt> students\_details = LOAD '/home/acadgild/assignment\_4.2/students\_details.txt' USING PigStorage(',') AS (roll\_no:int,name:chararray,place:chararray,age:int);

dump students\_details;

(1,Ram Sharma,Delhi,22)

(2,Shyam Kumar,Mumbai,28)

(3,Madhu Patel,Ahmedabad,29)

(4,Rohit Jain,Bangalore,25)

(5,Biju Nair,Kochi,25)

(5,Biju Nair,Kochi,25)

(6,Hari Sharma,Kolkata,25)

(6,Hari Sharma,Kolkata,25)

Next Load datasets from students\_marks.txt

Details of dataset students\_marks.txt is as below:

[acadgild@localhost assignment\_4.2]$ cat students\_marks.txt

1,Ram,Sharma,English,67

1,Ram,Sharma,Mathematics,85

1,Ram,Sharma,Science,65

2,Shyam,Kumar,English,48

2,Shyam,Kumar,Mathematics,82

2,Shyam,Kumar,Science,58

3,Madhu,Patel,English,71

3,Madhu,Patel,Mathematics,92

3,Madhu,Patel,Mathematics,78

4,Rohit,Jain,English,65

4,Rohit,Jain,Mathematics,81

4,Rohit,Jain,Mathematics,39

grunt> students\_marks = LOAD '/home/acadgild/assignment\_4.2/students\_marks.txt' USING PigStorage(',') AS (roll\_no:int,first\_name:chararray,last\_name:chararray,subject:chararray,marks:int);

grunt> dump students\_marks;

(1,Ram,Sharma,English,67)

(1,Ram,Sharma,Mathematics,85)

(1,Ram,Sharma,Science,65)

(2,Shyam,Kumar,English,48)

(2,Shyam,Kumar,Mathematics,82)

(2,Shyam,Kumar,Science,58)

(3,Madhu,Patel,English,71)

(3,Madhu,Patel,Mathematics,92)

(3,Madhu,Patel,Mathematics,78)

(4,Rohit,Jain,English,65)

(4,Rohit,Jain,Mathematics,81)

(4,Rohit,Jain,Mathematics,39)

1. Concat

Concat function is used to concatenation of two or more expression of

same type. Below, first\_name and last\_name are concatenated with '-'

first\_name\_last\_name = FOREACH students\_marks GENERATE CONCAT(first\_name, '-', last\_name);

dump first\_name\_last\_name;

(Ram-Sharma)

(Ram-Sharma)

(Ram-Sharma)

(Shyam-Kumar)

(Shyam-Kumar)

(Shyam-Kumar)

(Madhu-Patel)

(Madhu-Patel)

(Madhu-Patel)

(Rohit-Jain)

(Rohit-Jain)

(Rohit-Jain)

2. Tokenize

Tokenize is used to split a string in a single tuple returns a bag which

contains output of split opeartion. Below name is split using Tokenize pig operator

grunt> student\_name\_tokenize = FOREACH students\_details GENERATE TOKENIZE(name);

grunt> dump student\_name\_tokenize;

({(Ram),(Sharma)})

({(Shyam),(Kumar)})

({(Madhu),(Patel)})

({(Rohit),(Jain)})

({(Biju),(Nair)})

({(Biju),(Nair)})

({(Hari),(Sharma)})

({(Hari),(Sharma)})

3. Sum

Sum does total of numerical field of a column in a single bag. Below students marks are grouped by roll\_no and sum the marks for each roll\_no

grunt> students\_marks\_group\_by\_roll\_no = Group students\_marks BY roll\_no;

grunt> students\_marks\_total\_by\_roll\_no = FOREACH students\_marks\_group\_by\_roll\_no GENERATE group, SUM(students\_marks.marks);

grunt> dump students\_marks\_total\_by\_roll\_no;

(1,217)

(2,188)

(3,241)

(4,185)

4. Min

Min finds Minimum value for a column in a single bag. Below student marks are grouped by subject and for each subject minimum is calulcated using Min pig operator

grunt> students\_marks\_group\_subject = Group students\_marks BY subject;

grunt> students\_marks\_min\_by\_subject = FOREACH students\_marks\_group\_subject GENERATE group, MIN(students\_marks.marks);

grunt> dump students\_marks\_min\_by\_subject;

(English,48)

(Science,58)

(Mathematics,39)

5. Max

Max finds maximum value of a column in a single bag. Below student marks are grouped by subject and for each subject minimum is calulcated using Max pig operator

grunt> students\_marks\_group\_subject = Group students\_marks BY subject;

grunt> students\_marks\_max\_by\_subject = FOREACH students\_marks\_group\_subject GENERATE group, MAX(students\_marks.marks);

grunt> dump students\_marks\_max\_by\_subject;

(English,71)

(Science,65)

(Mathematics,92)

6. Limit

Limit function is used to get limited number of tuples. Below, 4 tuples of students\_details are retrieved using Limit function

grunt> students\_details\_limit = LIMIT students\_details 4;

grunt> dump students\_details\_limit;

(1,Ram Sharma,Delhi,22)

(2,Shyam Kumar,Mumbai,28)

(3,Madhu Patel,Ahmedabad,29)

(4,Rohit Jain,Bangalore,25)

7. Store

Store operator stores a loaded data into file system.

Below exmplae students\_details is stored to /home/acadgild/pig\_output

grunt> STORE students\_details INTO '/home/acadgild/pig\_output' USING PigStorage(',');

[acadgild@localhost ~]$ ls -l pig\_output/

total 4

-rw-r--r--. 1 acadgild acadgild 191 Oct 24 23:23 part-m-00000

-rw-r--r--. 1 acadgild acadgild 0 Oct 24 23:23 \_SUCCESS

[acadgild@localhost ~]$ cat pig\_output/part-m-00000

1,Ram Sharma,Delhi,22

2,Shyam Kumar,Mumbai,28

3,Madhu Patel,Ahmedabad,29

4,Rohit Jain,Bangalore,25

5,Biju Nair,Kochi,25

5,Biju Nair,Kochi,25

6,Hari Sharma,Kolkata,25

6,Hari Sharma,Kolkata,25

8. Distinct

Distinct finds the distinct tuples. Below, distict students details are returned

grunt> students\_details\_distinct = DISTINCT students\_details;

grunt> dump students\_details\_distinct;

(1,Ram Sharma,Delhi,22)

(2,Shyam Kumar,Mumbai,28)

(3,Madhu Patel,Ahmedabad,29)

(4,Rohit Jain,Bangalore,25)

(5,Biju Nair,Kochi,25)

(6,Hari Sharma,Kolkata,25)

9. Flatten

Flatten operator does unnesting of a bag or tuple. In this example below, first

create a cogroup students\_details\_marks using students\_details\_distinct and

students\_marks by roll\_no. Next I flatten bot students\_details\_distinct and

students\_marks

grunt> students\_details\_marks= COGROUP students\_details\_distinct by roll\_no, students\_marks by roll\_no;

grunt> dump students\_details\_marks;

(1,{(1,Ram Sharma,Delhi,22)},{(1,Ram,Sharma,Science,65),(1,Ram,Sharma,Mathematics,85),(1,Ram,Sharma,English,67)})

(2,{(2,Shyam Kumar,Mumbai,28)},{(2,Shyam,Kumar,Science,58),(2,Shyam,Kumar,Mathematics,82),(2,Shyam,Kumar,English,48)})

(3,{(3,Madhu Patel,Ahmedabad,29)},{(3,Madhu,Patel,Mathematics,78),(3,Madhu,Patel,Mathematics,92),(3,Madhu,Patel,English,71)})

(4,{(4,Rohit Jain,Bangalore,25)},{(4,Rohit,Jain,Mathematics,39),(4,Rohit,Jain,Mathematics,81),(4,Rohit,Jain,English,65)})

(5,{(5,Biju Nair,Kochi,25)},{})

(6,{(6,Hari Sharma,Kolkata,25)},{})

grunt> students\_details\_marks\_flatten = FOREACH students\_details\_marks GENERATE group, FLATTEN(students\_details\_distinct), FLATTEN(students\_marks);

(1,1,Ram Sharma,Delhi,22,1,Ram,Sharma,Science,65)

(1,1,Ram Sharma,Delhi,22,1,Ram,Sharma,Mathematics,85)

(1,1,Ram Sharma,Delhi,22,1,Ram,Sharma,English,67)

(2,2,Shyam Kumar,Mumbai,28,2,Shyam,Kumar,Science,58)

(2,2,Shyam Kumar,Mumbai,28,2,Shyam,Kumar,Mathematics,82)

(2,2,Shyam Kumar,Mumbai,28,2,Shyam,Kumar,English,48)

(3,3,Madhu Patel,Ahmedabad,29,3,Madhu,Patel,Mathematics,78)

(3,3,Madhu Patel,Ahmedabad,29,3,Madhu,Patel,Mathematics,92)

(3,3,Madhu Patel,Ahmedabad,29,3,Madhu,Patel,English,71)

(4,4,Rohit Jain,Bangalore,25,4,Rohit,Jain,Mathematics,39)

(4,4,Rohit Jain,Bangalore,25,4,Rohit,Jain,Mathematics,81)

(4,4,Rohit Jain,Bangalore,25,4,Rohit,Jain,English,65)

10. IsEmpty

isEmpty Function of Pig find if a bag or map is empty.

In this example below, first I create a cogroup students\_details\_marks using

students\_details\_distinct and students\_marks by roll\_no. Next I filter

records which is Empty for students\_marks.

grunt> students\_details\_marks= COGROUP students\_details\_distinct by roll\_no, students\_marks by roll\_no;

grunt> students\_details\_marks\_empty\_data = FILTER students\_details\_marks BY IsEmpty(students\_marks);

grunt> dump students\_details\_marks\_empty\_data;

(5,{(5,Biju Nair,Kochi,25)},{})

(6,{(6,Hari Sharma,Kolkata,25)},{})