**ASSIGNMENT 5.1**

PROBLEM STATEMENT:

**1)** **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).**

Query:

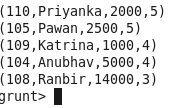
emp = LOAD '/home/acadgild/pig/employee\_details' USING PigStorage(',') AS (emp\_id:int, emp\_name:chararray, emp\_salary:int,emp\_ratings:int);

ordered\_emp = ORDER emp by emp\_ratings DESC;

five\_emp = LIMIT ordered\_emp 5;

dump five\_emp;

Output:



**2)** **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)**

Query:

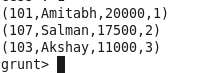
ordered\_emp\_salary = ORDER emp by emp\_salary DESC

filtered\_emp\_id = FILTER ordered\_emp\_salary by emp\_id%2!=0;

three\_emp = LIMIT filtered\_emp\_id 3;

dump three\_emp;

Output:



**3)** **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)**

Query:

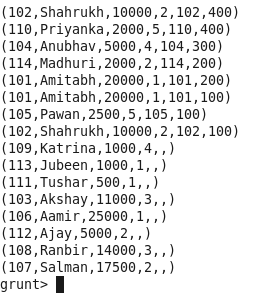
emp\_expenses = LOAD 'employee\_expenses' AS (emp\_id:int, expenses:int);

left\_outer\_joined\_data = join emp by emp\_id left outer, emp\_expenses by emp\_id;

ordered\_expenses = ORDER left\_outer\_joined\_data by emp\_expenses::expenses DESC;

dump ordered\_expenses;

Output:



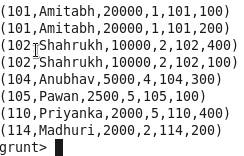
**4)** **List of employees (employee id and employee name) having entries in employee\_expenses file.**

Query:

filtered\_emp = FILTER left\_outer\_joined\_data by emp\_expenses::emp\_id is not null;

dump filtered\_emp;

Output:



**5)** **List of employees (employee id and employee name) having no entry in employee\_expenses file.**

Query:

filtered\_emp = FILTER left\_outer\_joined\_data by emp\_expenses::emp\_id is null;

Output:

