**Assignment 5.1**

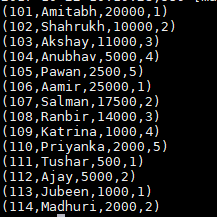
1. **Top 5 employees (employee id and employee name) with highest rating**. (In case two

employees have same rating, employee with name coming first in dictionary should get

preference)

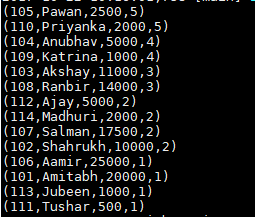
**grunt>** emp\_with\_rating = LOAD '/home/acadgild/pig/employee\_details.txt' USING PigStorage(',') AS (emp\_id:int, emp\_name:chararray, emp\_salary:int, emp\_rating:int);

**grunt>** DUMP emp\_with\_rating;



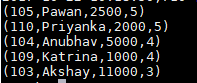
**grunt>** order\_emp\_name\_rating = ORDER emp\_with\_rating by emp\_rating desc, emp\_name asc;

**grunt>** DUMP order\_emp\_name\_rating;



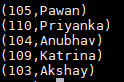
**grunt>** limit\_top\_5\_emp\_with\_rating =LIMIT order\_emp\_name\_rating 5;

**grunt>** DUMP limit\_top\_5\_emp\_with\_rating;



**grunt>** top\_5\_emp\_with\_rating = FOREACH limit\_top\_5\_emp\_with\_rating GENERATE emp\_id,emp\_name;

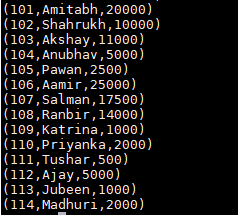
**grunt>** DUMP top\_5\_emp\_with\_rating;



1. **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)

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

**grunt>** DUMP emp;



**grunt>** emp\_odd\_even\_id = FOREACH emp GENERATE emp\_id,emp\_name,emp\_salary, (

>> CASE

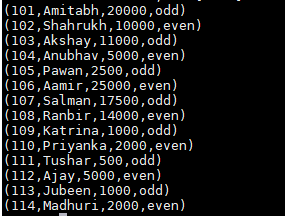
>> WHEN emp\_id % 2 == 0 THEN 'even'

>> WHEN emp\_id % 2 == 1 THEN 'odd'

>> END

>> ) AS odd\_even;

**grunt>** DUMPemp\_odd\_even\_id;



**grunt>** order\_emp\_odd\_even\_sal = ORDER emp\_odd\_even\_id by odd\_even desc,emp\_salary desc;

**grunt>**DUMP order\_emp\_odd\_even\_sal;



**grunt>** limit\_top\_3\_odd\_sal = LIMIT order\_emp\_odd\_even\_sal 3;

**grunt>**DUMP limit\_top\_3\_odd\_sal;



**grunt>** top\_3\_odd\_sal = FOREACH limit\_top\_3\_odd\_sal GENERATE emp\_id,emp\_name;

**grunt>** DUMPtop\_3\_odd\_sal;



1. **Employee (employee id and employee name) with maximum expense.** (In case two

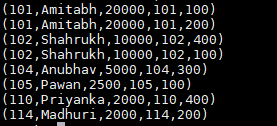
employees have same expense, employee with name coming first in dictionary should get preference)

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

**grunt>** emp\_expenses = LOAD '/home/acadgild/pig/employee\_expenses.txt' AS (emp\_id:int, expenses:int);

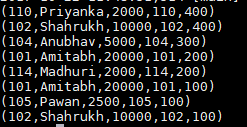
**grunt>** joined\_emp\_data = join emp by emp\_id , emp\_expenses by emp\_id;

**grunt>**DUMP joined\_emp\_data;



**grunt>** order\_emp\_expenses = ORDER joined\_data by emp\_expenses::expenses desc,emp::emp\_name asc;

**grunt>** DUMP order\_emp\_expenses;



**grunt>** limit\_emp\_with\_top\_expns = LIMIT order\_emp\_expenses 1;

**grunt>**DUMP limit\_emp\_with\_top\_expns;



**grunt>** emp\_with\_top\_expns = FOREACH emp\_with\_top\_expns GENERATE emp::emp\_id,emp::emp\_name;

**grunt>** DUMP emp\_with\_top\_expns;



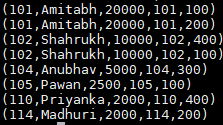
1. **List of employees (employee id and employee name) having entries in employee\_expenses file.**

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

**grunt>** emp\_expenses = LOAD '/home/acadgild/pig/employee\_expenses.txt' AS (emp\_id:int, expenses:int);

**grunt>** joined\_emp\_data = join emp by emp\_id , emp\_expenses by emp\_id;

**grunt>** DUMPjoined\_emp\_data;



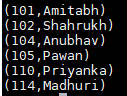
**grunt>** emp\_with\_entries\_expenses = FOREACH joined\_emp\_data GENERATE emp::emp\_id,emp::emp\_name;

**grunt>**  DUMP emp\_with\_entries\_expenses;



**grunt>** distinct\_emp\_with\_entries\_expenses = DISTINCT emp\_with\_entries\_expenses;

**grunt>**  DUMP distinct\_emp\_with\_entries\_expenses;



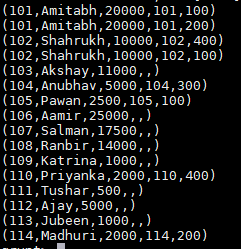
1. **List of employees (employee id and employee name) having no entry in employee\_expenses file.**

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

**grunt>** emp\_expenses = LOAD '/home/acadgild/pig/employee\_expenses.txt' AS (emp\_id:int, expenses:int);

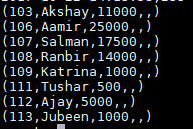
**grunt>** joined\_emp\_data = join emp by emp\_id left outer, emp\_expenses by emp\_id;

**grunt>** DUMP joined\_emp\_data;



**grunt>** filter\_emp\_expenses = FILTER joined\_emp\_data by emp\_expenses::emp\_id is null and emp\_expenses::expenses is null;

**grunt>** DUMP filter\_emp\_expenses;



**grunt>** emp\_not\_in\_expenses = FOREACH filter\_emp\_expenses generate $0,$1;

**grunt>** DUMP emp\_not\_in\_expenses;



**grunt>** distnct\_emp\_not\_in\_expenses = DISTINCT emp\_not\_in\_expenses;

**grunt>** dump distnct\_emp\_not\_in\_expenses;

