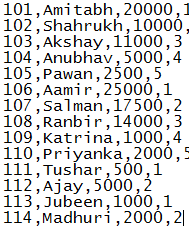
# **Input Data Sets:**

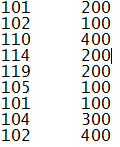
## *employee\_details.txt*:

<https://github.com/prateekATacadgild/DatasetsForCognizant/blob/master/employee_details.txt>



## employee\_expenses.txt:

[https://github.com/prateekATacadgild/DatasetsForCognizant/blob/master/employee\_expense s.txt](https://github.com/prateekATacadgild/DatasetsForCognizant/blob/master/employee_expense%20s.txt)



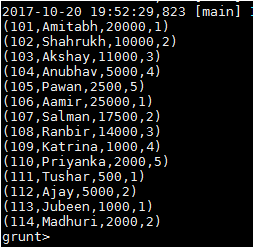
# **Load Data into Pig Storage:**

Step 1:-----Load the data sets to Pig Storage.

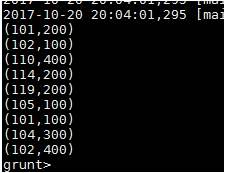
***employee\_details = LOAD '/home/acadgild/hadoop/employee\_details.txt' USING PigStorage(',') AS (id:int, name:chararray, salary:int, rating:int);***

***employee\_expenses = LOAD '/home/acadgild/hadoop/employee\_expenses.txt' USING PigStorage() AS (id:int, expenses:int);***

***DUMP employee\_details;***



***DUMP employee\_expenses;***



# Task 1:--

(a) 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)

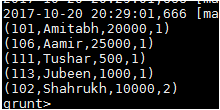
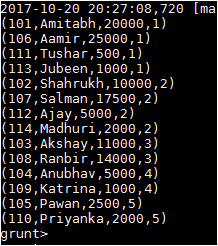
## **Answer:-**

The function ORDER to order the rating and the ID by ascending order and we are limiting the employee by top 5 with ids and the names

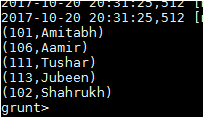
***employee = ORDER employee\_details BY rating asc, id asc;***

***limit\_employee = LIMIT employee 5;***

***top\_employee = FOREACH limit\_employee GENERATE id,name;***



## **Output**

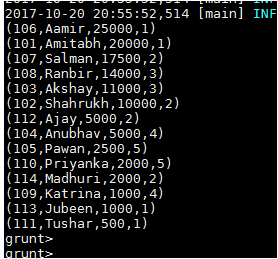
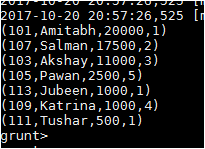
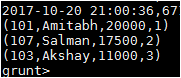


# **Task 2:---**

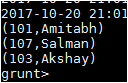
(b) Top 3 employees (employee id and employee name) with highest salary, whose employee idis an odd number. (In case two employees have same salary, employee with name coming firstin dictionary should get preference)

## **Answer:--**

## **T**he function ORDER to order the employee\_details by arranging the salary in descending order and filtering the even number using the logic **(id%2)!=0** and limiting the employee by 3 and bringing the result according to the employee ID and Name.

1. ***order\_employee = ORDER employee\_details BY salary DESC;***
2. ***filter\_employee = FILTER order\_employee BY (id%2)!=0;***
3. ***limit\_employee = LIMIT filter\_employee 3;***
4. ***top\_3employee = FOREACH limit\_employee GENERATE id,name;***
5. ***DUMP top\_3employee;***
6. 
7. 
8. 

## Output



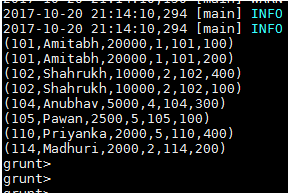
# **Task c**

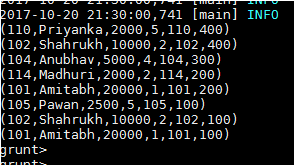
(c) Employee (employee id and employee name) with maximum expense (In case two employees have same expense, employee with name coming first in dictionary should get preference)

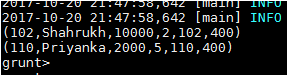
## **Answer:--**

Use joins-- the tables employee\_details and employee\_expenses using the JOIN function and sorting the expenses by Descending order and limiting the employee first 2 rows and bringing the id and name of the employee as the output.

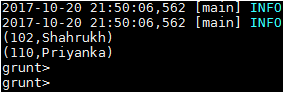
1. ***emp\_det\_exp = JOIN employee\_details BY id, emp\_expenses by id;***
2. ***order\_exp = ORDER emp\_det\_exp BY expenses DESC;***
3. ***limit\_exp = LIMIT order\_exp 2;***
4. ***max\_exp = FOREACH limit\_exp GENERATE $0,$1;***
5. ***DUMP max\_exp;***

1.

2.

3. 

## Output



# **Task 4**

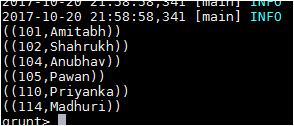
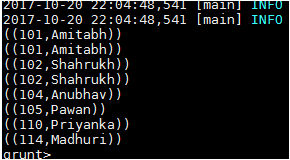
(d) List of employees (employee id and employee name) having entries in **employee\_expenses** file.

## **Answer:-**

## **J**oining the table’s employee\_details and employee\_expenses using the JOIN function and removing the redundant tuples from the relation and displaying the ID and name of the employee.

1. ***emp\_det\_exp = JOIN employee\_details BY id, emp\_expenses by id;***
2. ***emp\_names = FOREACH emp\_det\_exp GENERATE($0,$1);***
3. ***names = DISTINCT emp\_names;***
4. ***DUMP names;***

## Output



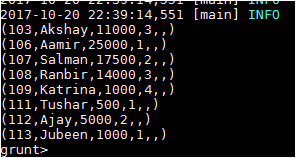
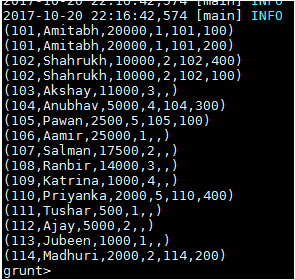
# **Task 5**

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

## Answer:---

Use the LEFT OUTER JOIN, we are joining the table’s employee\_details and employee\_expenses and filtering the empty values in the columns ($4: employee\_expenses.id and $5: employee\_expenses.expenses) and generating the output by ID and the name of the employee.

1. ***emp\_det\_exp = JOIN employee\_details BY id LEFT OUTER, employee\_expenses BY id;***
2. ***filter\_emp\_det\_exp = FILTER emp\_det\_exp BY $4 Is NULL and $5 Is NULL;***
3. ***gen\_emp\_det\_exp = FOREACH filter\_emp\_det\_exp GENERATE $0,$1;***
4. ***DUMP gen\_emp\_det\_exp;***



## Output

