## (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)

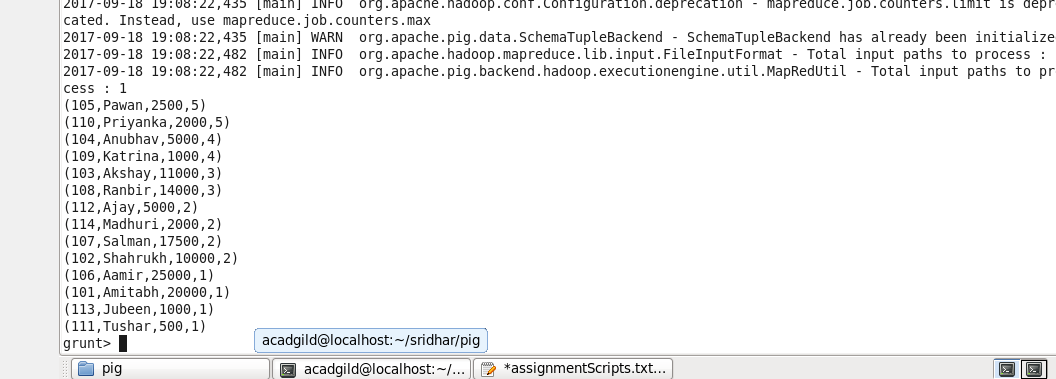
1. empDetails = LOAD '/home/acadgild/sridhar/pig/problem3/employee\_details.txt' USING PigStorage(',');

--Load the emp details file using PigStorage load func

1. empOrder = ORDER empDetails by $3 DESC , $1 ASC;

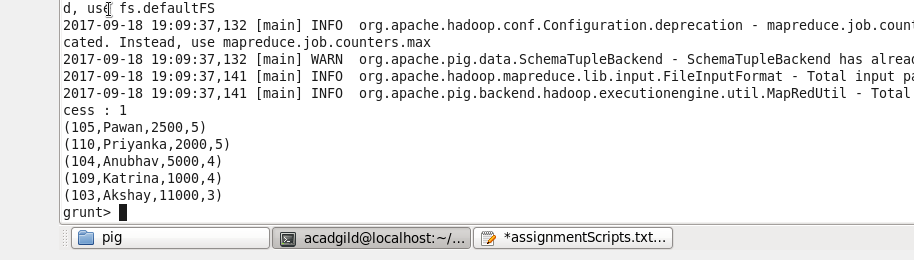
--sort the relation by rating and name

--the relation will be sorted by rating in descending order and when the rating is same, relation will sorted based on name in ascending order



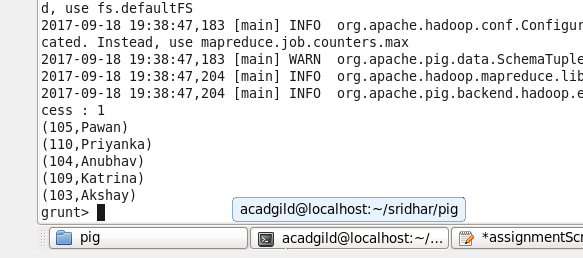
1. top5Emp = LIMIT empOrder 5 ;

Get the top 5 emp rating with using LIMIT operator



1. topEmpRating = FOREACH top5Emp GENERATE $0, $1; **(Final output)**

--fetch the emp id and name



## (b) 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)

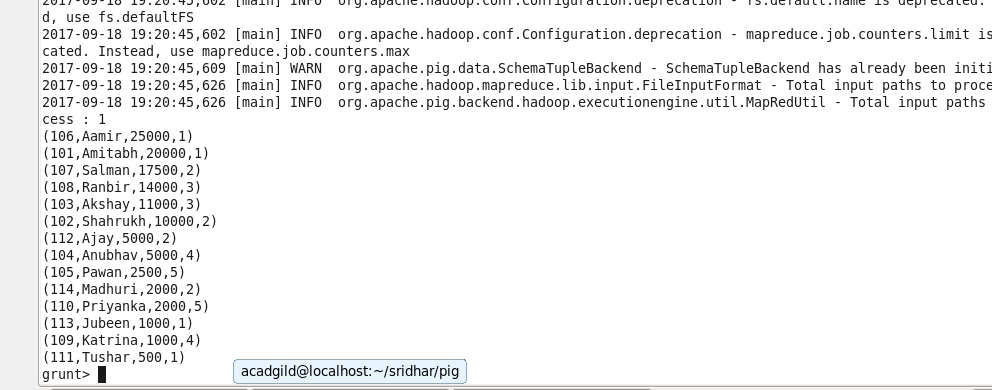
1. empDetails = LOAD '/home/acadgild/sridhar/pig/problem3/employee\_details.txt' USING PigStorage(',') as (id : int , name : chararray ,sal: int , rating : int);

--Load the emp details file using PigStorage load func

1. empOrderSal = ORDER empDetails by $2 desc , $1 ASC;

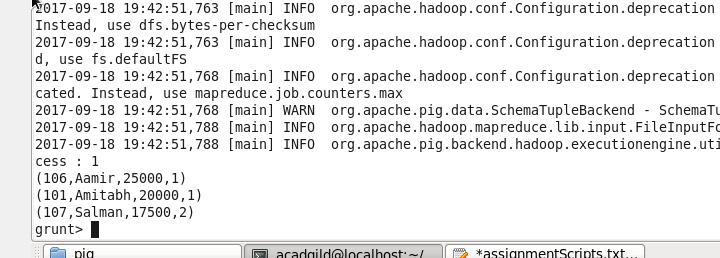
--sort the relation by salary and name

--the relation will be sorted by salary in descending order and when the salary is same, relation will sorted based on name in ascending order



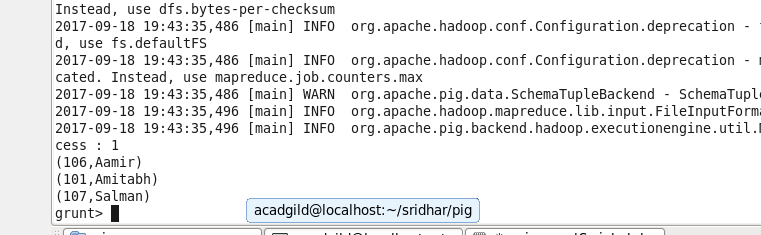
1. top3EmpSal = LIMIT empOrderSal 3;

-- Get the top 3 emp salary with using LIMIT operator



1. Top3EmpSalName = FOREACH top3EmpSal GENERATE $0 , $1; **(Final output)**

--Fetch the emp id and name



## (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)

1. empDetails = LOAD '/home/acadgild/sridhar/pig/problem3/employee\_details.txt' USING PigStorage(',') as (id : int , name : chararray ,sal: int , rating : int);

--Load the emp details file using PigStorage load func

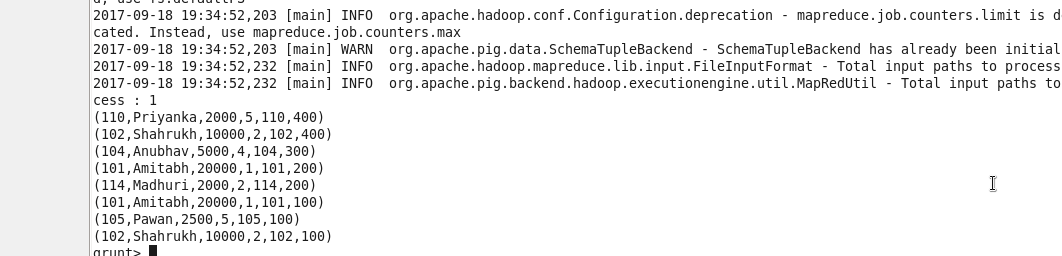
1. empExpense = LOAD '/home/acadgild/sridhar/pig/problem3/employee\_expenses.txt' USING PigStorage() as (id : int , exp: int);

--Load the emp expenses file using PigStorage load func

1. joinEmp = JOIN empDetails by id , empExpense by id;

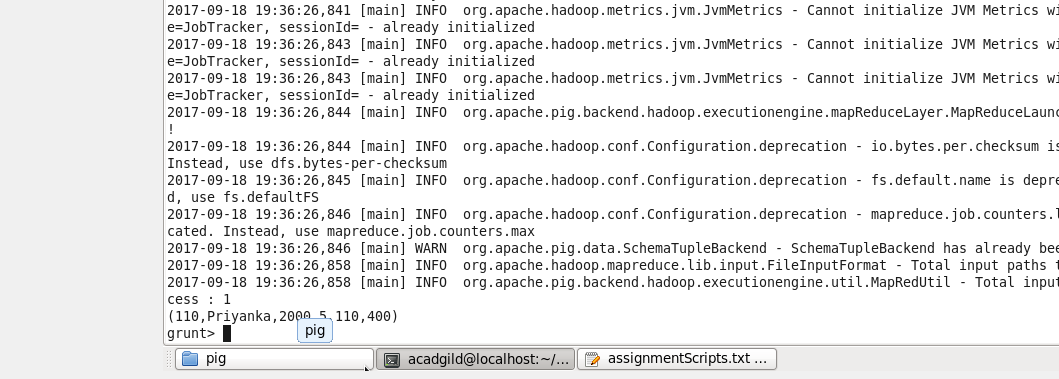
--Perform inner join so as to get the emp details those who have made expenses

1. orderEmpExp = ORDER joinEmp by $5 DESC , $1 ASC;



--Sort the relation based on highest expenses and and name in ascending order

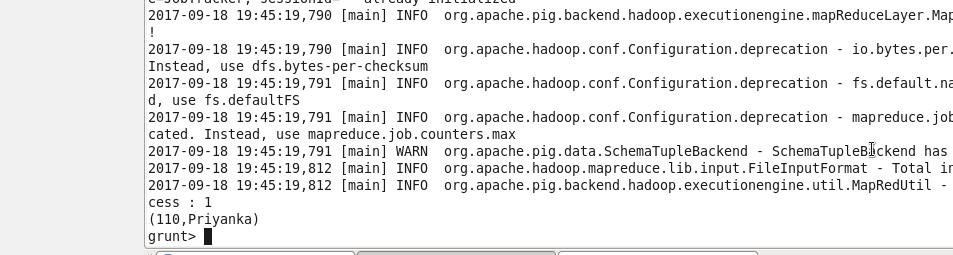
1. TopEmpExp = LIMIT orderEmpExp 1;



--Fetch the Employee with highest expense

1. TopEmpExpName = FOREACH TopEmpExp GENERATE $0 ,$1; **(Final output)**

--Generate emp id and name



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

1. empDetails = LOAD '/home/acadgild/sridhar/pig/problem3/employee\_details.txt' USING PigStorage(',') as (id : int , name : chararray ,sal: int , rating : int);

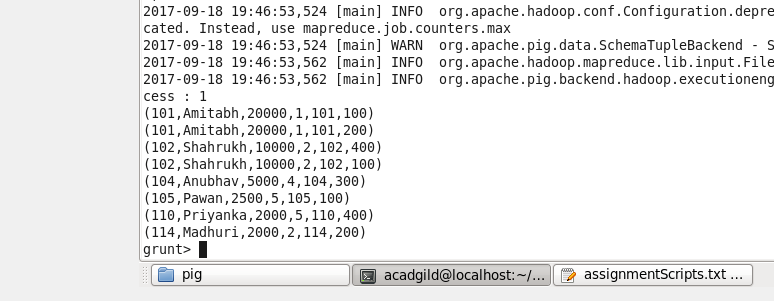
--Load the emp expenses file using PigStorage load func

1. empExpense = LOAD '/home/acadgild/sridhar/pig/problem3/employee\_expenses.txt' USING PigStorage() as (id : int , exp: int);

--Load the emp expenses file using PigStorage load func

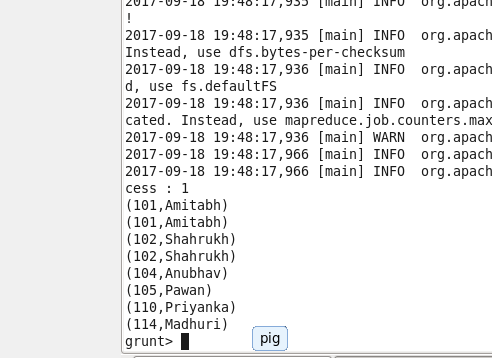
1. joinEmp = JOIN empDetails by id , empExpense by id;

--Perform inner join to get the employee who have entries in expenses relation



1. empExpEntries = FOREACH joinEmp GENERATE $0 , $1; **(Final output)**

--Generate emp id and emp name



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

1. empDetails = LOAD '/home/acadgild/sridhar/pig/problem3/employee\_details.txt' USING PigStorage(',') as (id : int , name : chararray ,sal: int , rating : int);

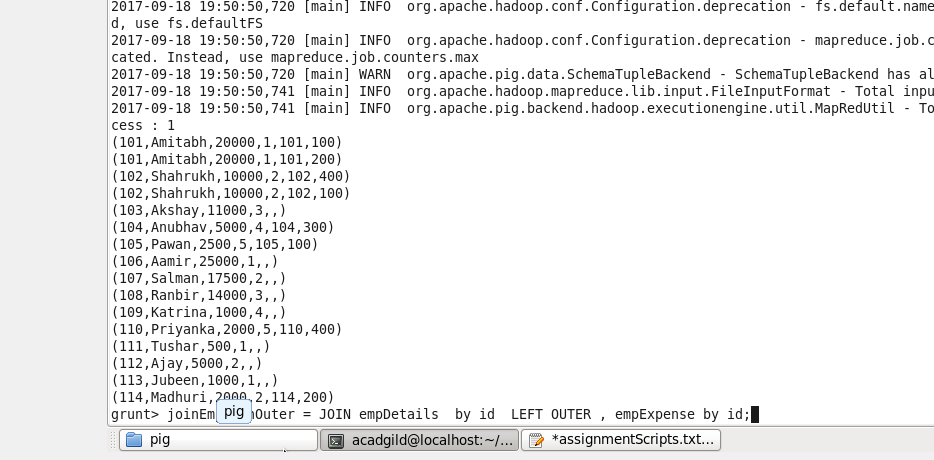
--Load the emp expenses file using PigStorage load func

1. empExpense = LOAD '/home/acadgild/sridhar/pig/problem3/employee\_expenses.txt' USING PigStorage() as (id : int , exp: int);

--Load the emp expenses file using PigStorage load func

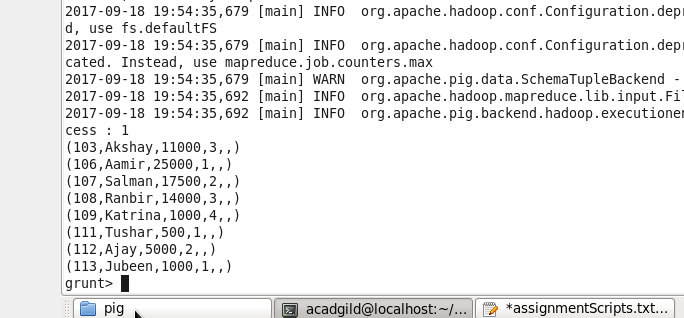
1. joinEmpRighOuter = JOIN empDetails by id LEFT OUTER , empExpense by id;

--Perform left inner join to get the employee who do have entries in expenses relation



1. empExpNoEntries = FILTER joinEmpRighOuter by $4 is null ;

--filter emp with no expenses



1. empExpNoEntriesName = FOREACH empExpNoEntries GENERATE $0 ,$1; **(Final output)**

--Fetch emp id and emp name

