**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)

**PIG LATIN SCRIPT:**

*emp\_group = GROUP emp\_details by rating;*

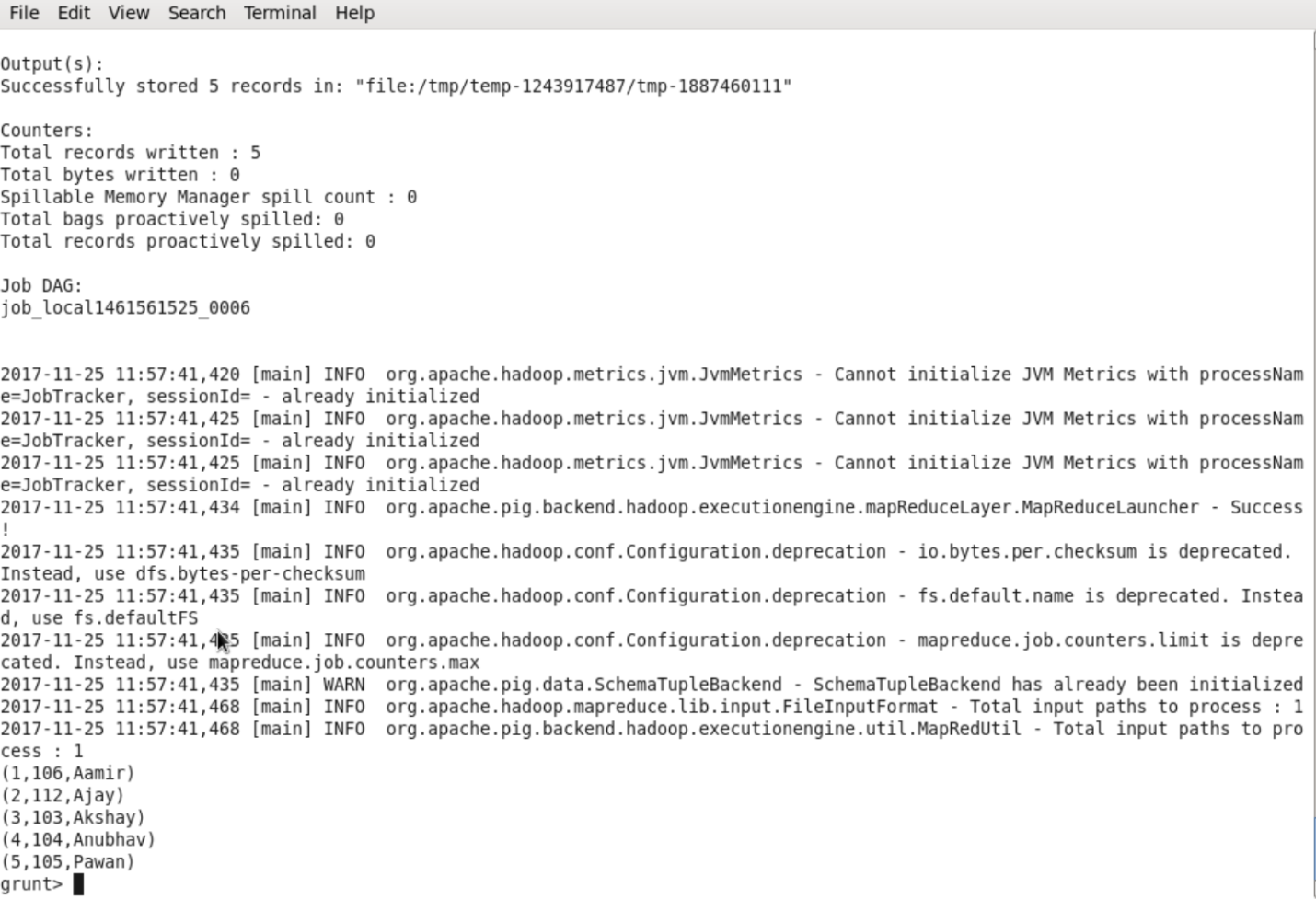
*emp\_5 = FOREACH emp\_group {*

*da = ORDER emp\_details BY rating DESC,name;*

*db = LIMIT da 1;*

*GENERATE FLATTEN(group), FLATTEN(db.id), FLATTEN(db.name);*

*}*



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)

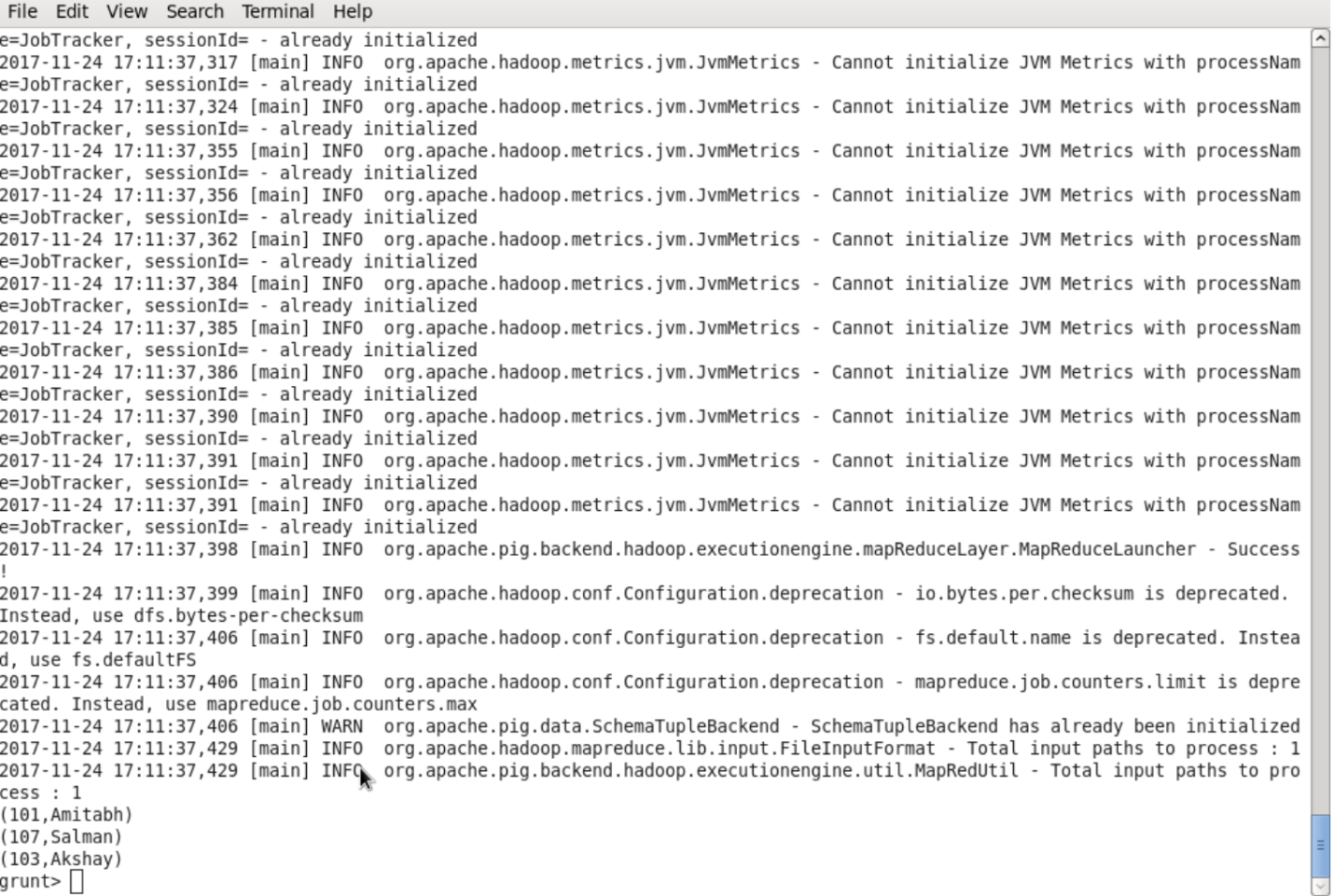
**PIG LATIN SCRIPT:**

*emp\_odd = FILTER emp\_details BY (id%2 !=0);*

*emp\_odd\_order = ORDER emp\_odd BY salary desc, name;*

*emp = LIMIT emp\_odd\_order 3;*

*emp\_result = FOREACH emp GENERATE id, name;*



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).

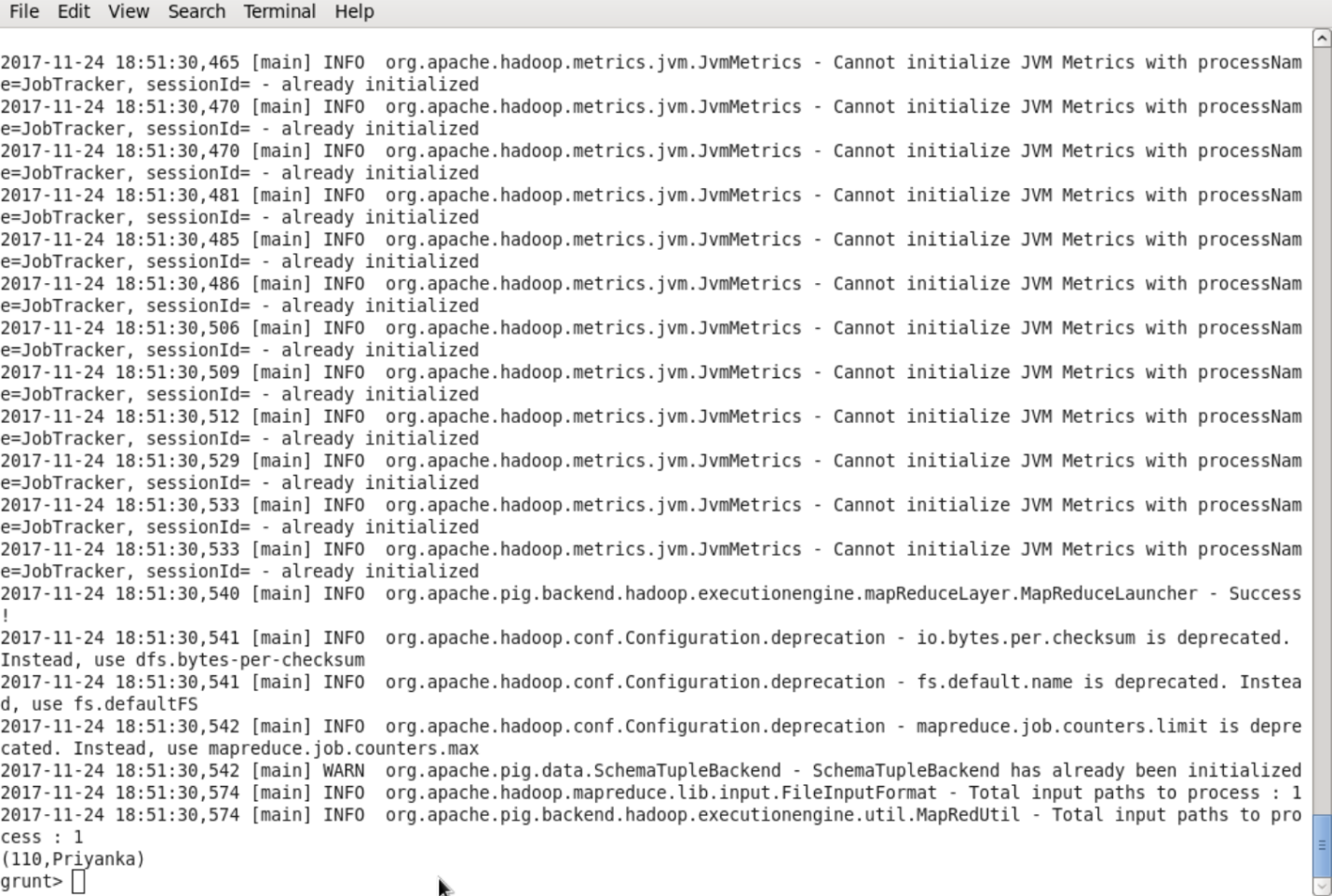
**PIG LATIN SCRIPT:**

*A = JOIN emp\_details BY id, emp\_expenses BY id;*

*B = ORDER A BY expense DESC, name;*

*C = LIMIT B 1;*

*D = FOREACH C GENERATE emp\_details::id, name;*



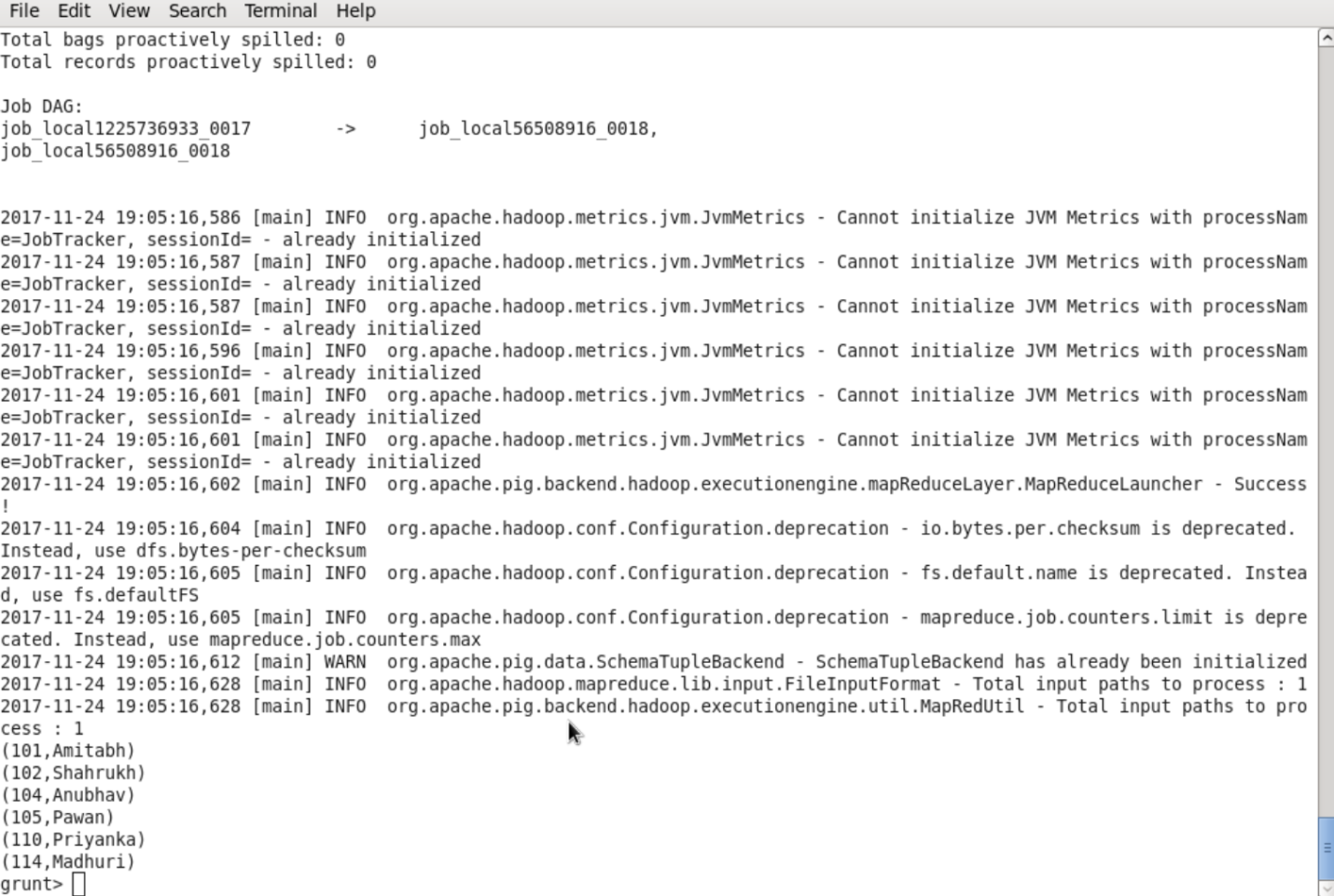
1. List of employees (employee id and employee name) having entries in employee\_expenses file.

**PIG LATIN SCRIPT:**

*joined = JOIN emp\_details BY id, emp\_expenses BY id;*

*emp = FOREACH joined GENERATE emp\_details::id, name;*

*emp\_data = DISTINCT emp;*



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

**PIG LATIN SCRIPT:**

*joined = JOIN emp\_details BY id, emp\_expenses BY id;*

*a = FILTER joined BY IsEmpty(emp\_expenses);*

*b = FOREACH a GENERATE FLATTEN(emp\_details.id), FLATTEN(emp\_details.name);*

