

Assignment 7.2:

Problem 7.2.a:

```
1.) emp = LOAD 'employee_details' USING PigStorage(',') AS (emp_id:int, emp_name:chararray,
emp_salary:int);
    dump emp;
2.)top_emp = ORDER emp by emp_rating desc;
    dump top_emp;
3.)top_five_emp = limit top_emp 5;
    dump top_five_emp;
4.)top_five_emp_name = FOREACH top_five_emp generate emp_id, emp_name;
    dump top_five_emp_name;
```

Problem 7.2.b

```
1.) emp = LOAD 'employee_details' USING PigStorage(',') AS (emp_id:int, emp_name:chararray,
emp_salary:int);
    dump emp;
2.)top_emp = ORDER emp by emp_salary desc;
    dump top_emp;
3.)top_odd_emp = FILTER top_emp BY emp_id%2!=0;
    dump top_odd_emp;
4.)top_three_emp = limit top_odd_emp 3;
    dump top_three_emp;
5.)top_three_emp_name = FOREACH top_three_emp generate emp_id, emp_name;
    dump top_three_emp_name;
```

Problem 7.2.C

```
1.) emp = LOAD 'employee_details' USING PigStorage(',') AS (emp_id:int, emp_name:chararray);
    dump emp;
2.) emp_expenses = LOAD 'employee_expenses' AS (emp_id:int, expenses:int);
    dump emp_expenses;
3.) grouped_expense = group emp_expenses by emp_id;
    dump grouped_expense;
4.) grouped_expense_sum = foreach grouped_expense generate group as emp_id, SUM(emp_expenses.expenses) as
total_expense;
    dump grouped_expense_sum;
5.) sorted_expenses = ORDER grouped_expense_sum BY total_expense desc;
    dump sorted_expenses;
6.) top_expense_emp = limit sorted_expenses 1;
    dump top_expense_emp;
7.) top_exp_expenses = FILTER emp by (emp_id==102);
    dump top_expense_emp;
```

Problem 7.2.D

```
1.) joined_data = join emp by emp_id , emp_expenses by emp_id;
    dump joined_data;
    describe joined_data;
2.) emp_name_expense = FOREACH joined_data generate emp::emp_id,emp::emp_name;
    dump emp_name_expense;
3.) emp_name_expense_exist = DISTINCT emp_name_expense;
    dump emp_name_expense_exist;
```

Problem 7.2.E

```
1.)left_outer_joined_data = join emp by emp_id left outer, emp_expenses by emp_id;
    dump left_outer_joined_data;
    describe left_outer_joined_data;
2.) SPLIT left_outer_joined_data INTO emp_expense_exis IF (emp_expenses::expenses is not null),
```

```
emp_no_expense IF (emp_expenses::expenses is null);  
    dump emp_expense_exis;  
    dump emp_no_expense ;  
3.) emp_no_expense_name = FOREACH emp_no_expense generate emp::emp_id, emp::emp_name;  
    dump emp_no_expense_name;
```