

Task 2

```
[acadgild@localhost ASSIGNMENT]$ ls ./
employee_details.txt  employee_expenses.txt
[acadgild@localhost ASSIGNMENT]$
```

```
[acadgild@localhost ASSIGNMENT]$ pig -brief -x local
```

(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)

```
grunt> employee = LOAD './employee_details.txt' USING PigStorage(',') as (id:int,name:chararray,salary:int,rating:int);
grunt> dump employee;
```

```
(101,Amitabh,20000,1)
(102,Shahrukh,10000,2)
(103,Akshay,11000,3)
(104,Anubhav,5000,4)
(105,Pawan,2500,5)
(106,Aamir,25000,1)
(107,Salman,17500,2)
(108,Ranbir,14000,3)
(109,Katrina,1000,4)
(110,Priyanka,2000,5)
(111,Tushar,500,1)
(112,Ajay,5000,2)
(113,Jubeen,1000,1)
(114,Madhuri,2000,2)
```

```
grunt> employee_order = ORDER employee by rating desc,name asc;
grunt> employee_ord_limit = LIMIT employee_order 5;
grunt> dump employee_ord_limit;
```

```
(105,Pawan,2500,5)
(110,Priyanka,2000,5)
(104,Anubhav,5000,4)
(109,Katrina,1000,4)
(103,Akshay,11000,3)
grunt>
```

```
grunt> employee_out = foreach employee_ord_limit generate id,name;
grunt> dump employee_out;
```

```
(105,Pawan)
(110,Priyanka)
(104,Anubhav)
(109,Katrina)
(103,Akshay)
```

(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)

```
grunt> employee = LOAD './employee_details.txt' USING PigStorage(',') as (id:int,name:chararray,salary:int,rating:int);
grunt> dump employee;
```

```
(101,Amitabh,20000,1)
(102,Shahrukh,10000,2)
(103,Akshay,11000,3)
(104,Anubhav,5000,4)
(105,Pawan,2500,5)
(106,Aamir,25000,1)
(107,Salman,17500,2)
(108,Ranbir,14000,3)
(109,Katrina,1000,4)
(110,Priyanka,2000,5)
(111,Tushar,500,1)
(112,Ajay,5000,2)
(113,Jubeen,1000,1)
(114,Madhuri,2000,2)
```

```
grunt>employee_order_sal = ORDER employee by salary desc,name asc;
grunt>employee_odd = FILTER employee_order_sal BY (id % 2 != 0);
grunt> dump employee_odd;
```

```
Total input paths to process : 1
(101,Amitabh,20000,1)
(107,Salman,17500,2)
(103,Akshay,11000,3)
(105,Pawan,2500,5)
(113,Jubeen,1000,1)
(109,Katrina,1000,4)
(111,Tushar,500,1)
grunt>
```

```
grunt>employee_top_3_sal = LIMIT employee_odd 3;
grunt> DUMP employee_top_3_sal;
```

```
(101,Amitabh,20000,1)
(107,Salman,17500,2)
(103,Akshay,11000,3)
grunt>
```

```
grunt>top_3_salary = FOREACH employee_top_3_sal GENERATE id,name;
grunt> dump top_3_salary;
```

```
Total input paths to process : 1
Total input paths to process : 1
(101,Amitabh)
(107,Salman)
(103,Akshay)
```

(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)

```
grunt> employee_expenses = LOAD './employee_expenses.txt' as (id:int,expense:int);
grunt> dump employee_expenses;
```

```
(101,200)
(102,100)
(110,400)
(114,200)
(119,200)
(105,100)
(101,100)
(104,300)
(102,400)
```

```
grunt> employee = LOAD './employee_details.txt' USING PigStorage(',') as (id:int,name:chararray,salary:int,rating:int);
grunt> dump employee;
```

```
(101,Amitabh,20000,1)
(102,Shahrukh,10000,2)
(103,Akshay,11000,3)
(104,Anubhav,5000,4)
(105,Pawan,2500,5)
(106,Aamir,25000,1)
(107,Salman,17500,2)
(108,Ranbir,14000,3)
(109,Katrina,1000,4)
(110,Priyanka,2000,5)
(111,Tushar,500,1)
(112,Ajay,5000,2)
(113,Jubeen,1000,1)
(114,Madhuri,2000,2)
```

```
grunt> emp_exp_join = JOIN employee by id,employee_expenses by id;
grunt> dump emp_exp_join;
```

```
(101,Amitabh,20000,1,101,100)
(101,Amitabh,20000,1,101,200)
(102,Shahrukh,10000,2,102,400)
(102,Shahrukh,10000,2,102,100)
(104,Anubhav,5000,4,104,300)
(105,Pawan,2500,5,105,100)
(110,Priyanka,2000,5,110,400)
(114,Madhuri,2000,2,114,200)
```

```
grunt> emp_exp_srt = ORDER emp_exp_join by expense desc,name asc;
grunt> DUMP emp_exp_srt ;
```

```
(110,Priyanka,2000,5,110,400)
(102,Shahrukh,10000,2,102,400)
(104,Anubhav,5000,4,104,300)
(101,Amitabh,20000,1,101,200)
(114,Madhuri,2000,2,114,200)
(101,Amitabh,20000,1,101,100)
(105,Pawan,2500,5,105,100)
(102,Shahrukh,10000,2,102,100)
```

```
grunt> emp_expense_grp = GROUP emp_exp_srt by employee::id;
```

```
(101,{(101,Amitabh,20000,1,101,100),(101,Amitabh,20000,1,101,200)})
(102,{(102,Shahrukh,10000,2,102,100),(102,Shahrukh,10000,2,102,400)})
(104,{(104,Anubhav,5000,4,104,300)})
(105,{(105,Pawan,2500,5,105,100)})
(110,{(110,Priyanka,2000,5,110,400)})
(114,{(114,Madhuri,2000,2,114,200)})
```

```
grunt> emp_tot_expense = FOREACH emp_expense_grp GENERATE group as id,emp_exp_srt.name as
name,SUM(emp_exp_srt.expense) as expense;
grunt> dump emp_tot_expense;
```

```
(101,{(Amitabh),(Amitabh)},300)
(102,{(Shahrukh),(Shahrukh)},500)
(104,{(Anubhav)},300)
(105,{(Pawan)},100)
(110,{(Priyanka)},400)
(114,{(Madhuri)},200)
```

```
grunt> emp_tot_expense = FOREACH emp_expense_grp GENERATE group as id,emp_exp_srt.name as
name,SUM(emp_exp_srt.expense) as expense;
grunt> dump emp_tot_expense;
```

```
(101,Amitabh,300)
(101,Amitabh,300)
(102,Shahrukh,500)
(102,Shahrukh,500)
(104,Anubhav,300)
(105,Pawan,100)
(110,Priyanka,400)
(114,Madhuri,200)
```

```
grunt> emp_expense_details_unique = DISTINCT emp_expense_details;
grunt> dump emp_expense_details_unique;
```

```
(101,Amitabh,300)
(102,Shahrukh,500)
(104,Anubhav,300)
(105,Pawan,100)
(110,Priyanka,400)
(114,Madhuri,200)
```

```
grunt> emp_expense_sort_desc = ORDER emp_expense_details_unique by expense desc;
grunt> dump emp_expense_sort_desc;
```

```
(102,Shahrukh,500)
(110,Priyanka,400)
(104,Anubhav,300)
(101,Amitabh,300)
(114,Madhuri,200)
(105,Pawan,100)
```

```
grunt> employee_expense_out = FOREACH emp_expense_sort_desc GENERATE id,name;
grunt> dump employee_expense_out;
```

```
(102,Shahrukh)
(110,Priyanka)
(104,Anubhav)
(101,Amitabh)
(114,Madhuri)
(105,Pawan)
grunt>
```

(d) List of employees (employee id and employee name) having entries in employee_expenses file.

```
grunt> employees_in_expense = FOREACH employee_expense_out GENERATE id,name;
grunt> dump employees_in_expense;
```

```
(102,Shahrukh)
(110,Priyanka)
(104,Anubhav)
(101,Amitabh)
(114,Madhuri)
(105,Pawan)
```

(e) List of employees (employee id and employee name) having no entry in employee_expenses file

```
grunt> emp = LOAD './employee_details.txt' USING PigStorage(',') as (id:int,name:chararray,salary:int,rating:int);
```

```
grunt> emp_exp = LOAD './employee_expenses.txt' as (id:int,expense:int);
```

```
grunt> emp_join = JOIN emp by id LEFT,emp_exp by id;
```

```
grunt> dump emp_join;
```

```
(101,Amitabh,20000,1,101,100)
(101,Amitabh,20000,1,101,200)
(102,Shahrukh,10000,2,102,400)
(102,Shahrukh,10000,2,102,100)
(103,Akshay,11000,3,,)
(104,Anubhav,5000,4,104,300)
(105,Pawan,2500,5,105,100)
(106,Aamir,25000,1,,)
(107,Salman,17500,2,,)
(108,Ranbir,14000,3,,)
(109,Katrina,1000,4,,)
(110,Priyanka,2000,5,110,400)
(111,Tushar,500,1,,)
(112,Ajay,5000,2,,)
(113,Jubeen,1000,1,,)
(114,Madhuri,2000,2,114,200)
```

```
grunt> emp_for = FOREACH emp_join GENERATE emp::id,emp::name,emp_exp::expense;
```

```
grunt> dump emp_for;
```

```
(101,Amitabh,100)
(101,Amitabh,200)
(102,Shahrukh,400)
(102,Shahrukh,100)
(103,Akshay,)
(104,Anubhav,300)
(105,Pawan,100)
(106,Aamir,)
(107,Salman,)
(108,Ranbir,)
(109,Katrina,)
(110,Priyanka,400)
(111,Tushar,)
(112,Ajay,)
(113,Jubeen,)
(114,Madhuri,200)
```

```
grunt> emp_srt = ORDER emp_for by emp_exp::expense;
```

```
grunt> dump emp_srt;
```

```
(106,Aamir,)  
(113,Jubeen,)  
(112,Ajay,)  
(111,Tushar,)  
(103,Akshay,)  
(109,Katrina,)  
(108,Ranbir,)  
(107,Salman,)  
(101,Amitabh,100)  
(105,Pawan,100)  
(102,Shahrukh,100)  
(114,Madhuri,200)  
(101,Amitabh,200)  
(104,Anubhav,300)  
(110,Priyanka,400)  
(102,Shahrukh,400)
```

```
grunt> emp_temp_srt = FILTER emp_srt by emp_exp::expense is NULL;  
grunt> dump emp_temp_srt;
```

```
(113,Jubeen,)  
(112,Ajay,)  
(111,Tushar,)  
(109,Katrina,)  
(108,Ranbir,)  
(107,Salman,)  
(106,Aamir,)  
(103,Akshay,)  
grunt>
```

```
grunt> emp_op = FOREACH emp_temp_srt GENERATE id,name;  
grunt> dump emp_op;
```

```
(113,Jubeen)  
(112,Ajay)  
(111,Tushar)  
(109,Katrina)  
(108,Ranbir)  
(107,Salman)  
(106,Aamir)  
(103,Akshay)
```