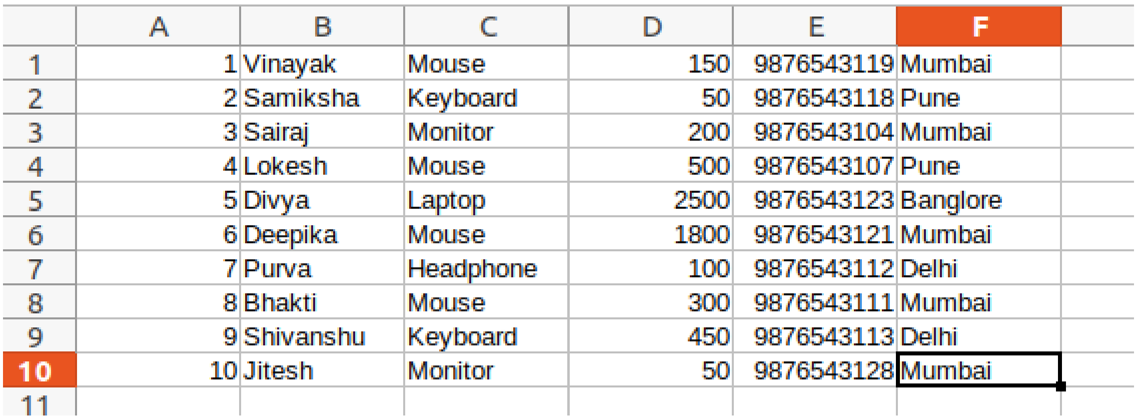
PIG Assignment

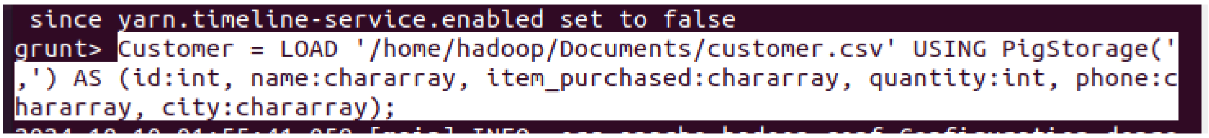
Q1) Perform the following PIG

1. **Create a .csv file to store customer details- id,name ,item_purchased, quantity,phone,city.**



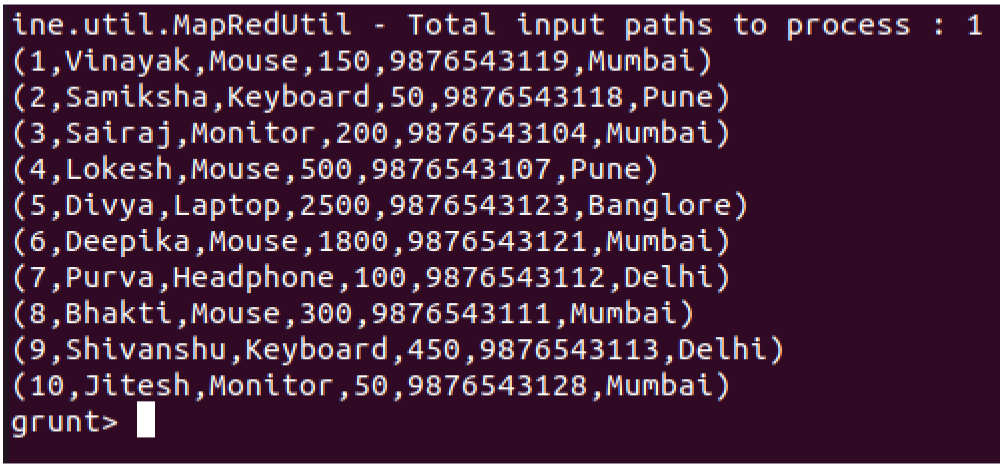
	A	B	C	D	E	F
1	1	Vinayak	Mouse	150	9876543119	Mumbai
2	2	Samiksha	Keyboard	50	9876543118	Pune
3	3	Sairaj	Monitor	200	9876543104	Mumbai
4	4	Lokesh	Mouse	500	9876543107	Pune
5	5	Divya	Laptop	2500	9876543123	Banglore
6	6	Deepika	Mouse	1800	9876543121	Mumbai
7	7	Purva	Headphone	100	9876543112	Delhi
8	8	Bhakti	Mouse	300	9876543111	Mumbai
9	9	Shivanshu	Keyboard	450	9876543113	Delhi
10	10	Jitesh	Monitor	50	9876543128	Mumbai
11						

2. **Create a relation CUSTOMER to store the details of this .csv file**



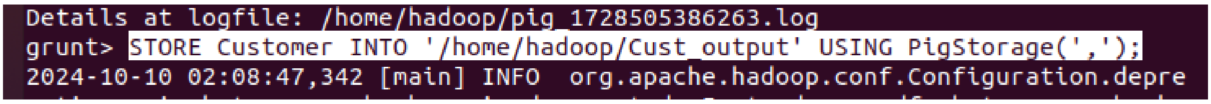
```
 since yarn.timeline-service.enabled set to false
grunt> Customer = LOAD '/home/hadoop/Documents/customer.csv' USING PigStorage('
,') AS (id:int, name:chararray, item_purchased:chararray, quantity:int, phone:c
hararray, city:chararray);
```

3. **Display the contents of this relation on screen**

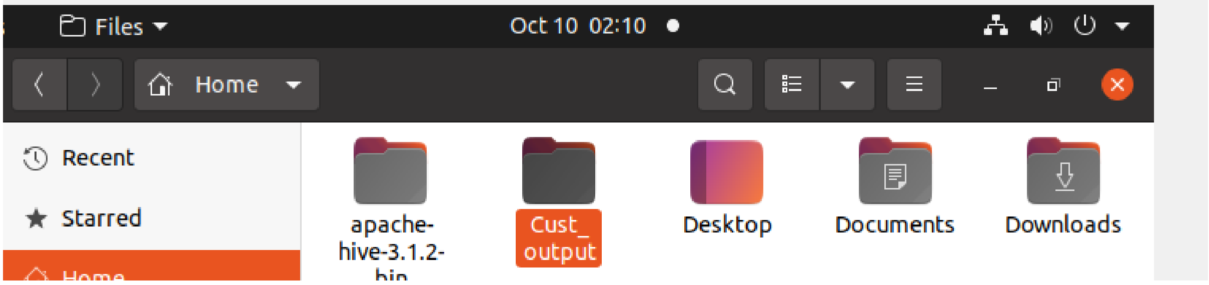


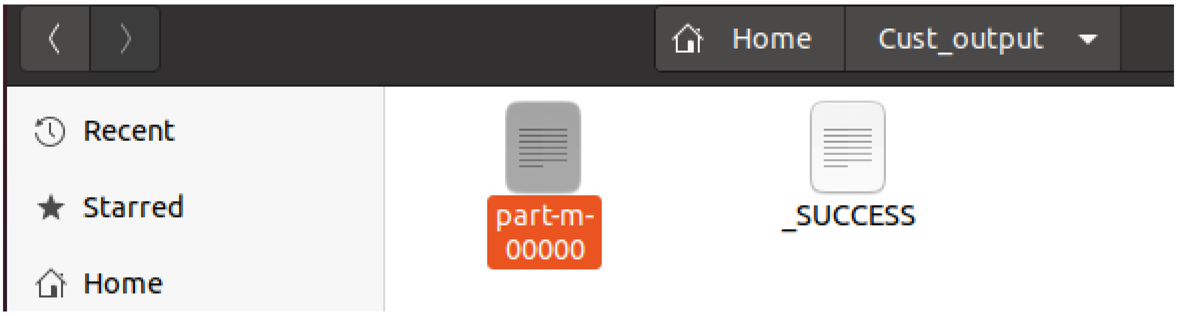
```
ine.util.MapRedUtil - Total input paths to process : 1
(1,Vinayak,Mouse,150,9876543119,Mumbai)
(2,Samiksha,Keyboard,50,9876543118,Pune)
(3,Sairaj,Monitor,200,9876543104,Mumbai)
(4,Lokesh,Mouse,500,9876543107,Pune)
(5,Divya,Laptop,2500,9876543123,Banglore)
(6,Deepika,Mouse,1800,9876543121,Mumbai)
(7,Purva,Headphone,100,9876543112,Delhi)
(8,Bhakti,Mouse,300,9876543111,Mumbai)
(9,Shivanshu,Keyboard,450,9876543113,Delhi)
(10,Jitesh,Monitor,50,9876543128,Mumbai)
grunt> █
```

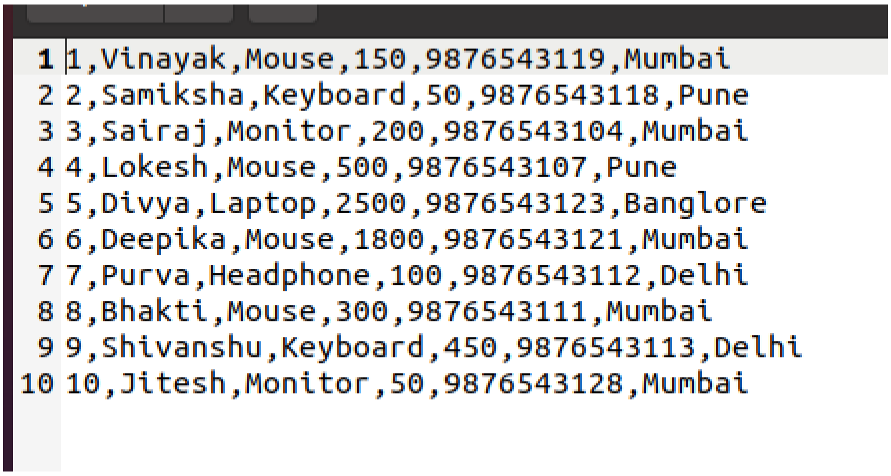
4. **Store this relation data in local file system**



```
 Details at logfile: /home/hadoop/pig_1728505386263.log
 grunt> STORE Customer INTO '/home/hadoop/Cust_output' USING PigStorage(',');
 2024-10-10 02:08:47,342 [main] INFO  org.apache.hadoop.conf.Configuration.depre
```

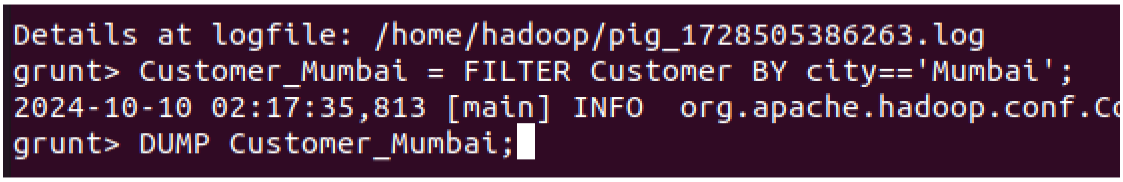




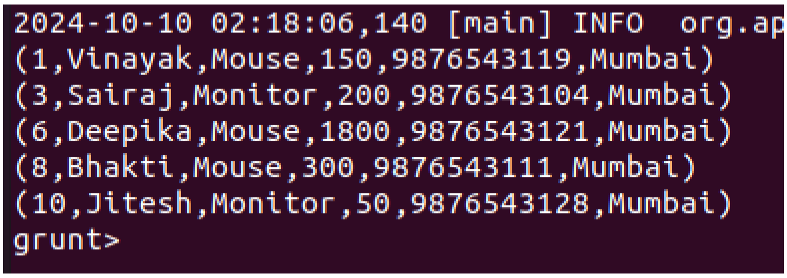


```
1,Vinayak,Mouse,150,9876543119,Mumbai
2,Samiksha,Keyboard,50,9876543118,Pune
3,Sairaj,Monitor,200,9876543104,Mumbai
4,Lokesh,Mouse,500,9876543107,Pune
5,Divya,Laptop,2500,9876543123,Banglore
6,Deepika,Mouse,1800,9876543121,Mumbai
7,Purva,Headphone,100,9876543112,Delhi
8,Bhakti,Mouse,300,9876543111,Mumbai
9,Shivanshu,Keyboard,450,9876543113,Delhi
10,Jitesh,Monitor,50,9876543128,Mumbai
```

5. Display details of customers whose city is ‘Mumbai’;

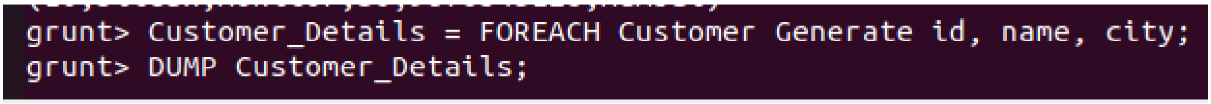


```
Details at logfile: /home/hadoop/pig_1728505386263.log
grunt> Customer_Mumbai = FILTER Customer BY city=='Mumbai';
2024-10-10 02:17:35,813 [main] INFO  org.apache.hadoop.conf.Co
grunt> DUMP Customer_Mumbai;
```

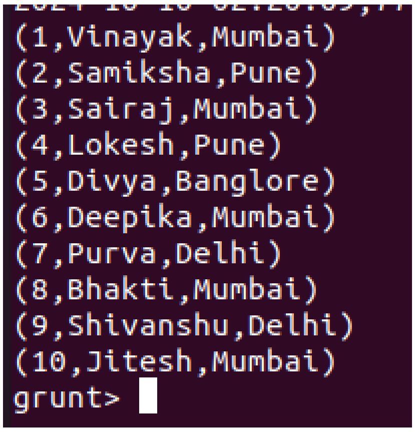


```
2024-10-10 02:18:06,140 [main] INFO  org.ap
(1,Vinayak,Mouse,150,9876543119,Mumbai)
(3,Sairaj,Monitor,200,9876543104,Mumbai)
(6,Deepika,Mouse,1800,9876543121,Mumbai)
(8,Bhakti,Mouse,300,9876543111,Mumbai)
(10,Jitesh,Monitor,50,9876543128,Mumbai)
grunt>
```

6. **Display id, name and city of all customers.**

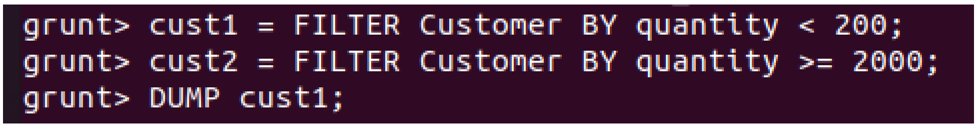


```
grunt> Customer_Details = FOREACH Customer Generate id, name, city;
grunt> DUMP Customer_Details;
```

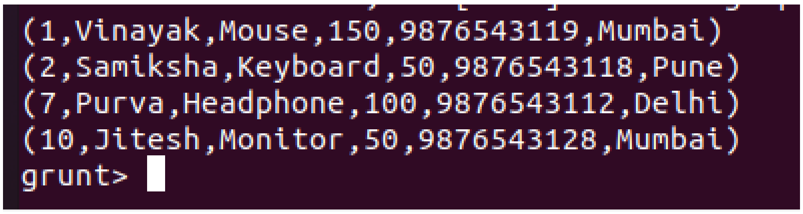


```
(1,Vinayak,Mumbai)
(2,Samiksha,Pune)
(3,Sairaj,Mumbai)
(4,Lokesh,Pune)
(5,Divya,Banglore)
(6,Deepika,Mumbai)
(7,Purva,Delhi)
(8,Bhakti,Mumbai)
(9,Shivanshu,Delhi)
(10,Jitesh,Mumbai)
grunt>
```

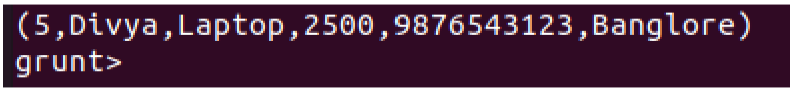
7. **Separate the contents of customer relation for quantity < 200 and >= 2000 to cust1 and cust2 respectively.**



```
grunt> cust1 = FILTER Customer BY quantity < 200;
grunt> cust2 = FILTER Customer BY quantity >= 2000;
grunt> DUMP cust1;
```

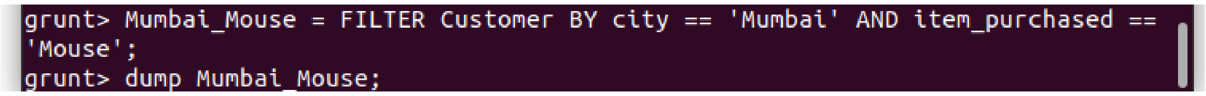


```
(1,Vinayak,Mouse,150,9876543119,Mumbai)
(2,Samiksha,Keyboard,50,9876543118,Pune)
(7,Purva,Headphone,100,9876543112,Delhi)
(10,Jitesh,Monitor,50,9876543128,Mumbai)
grunt>
```

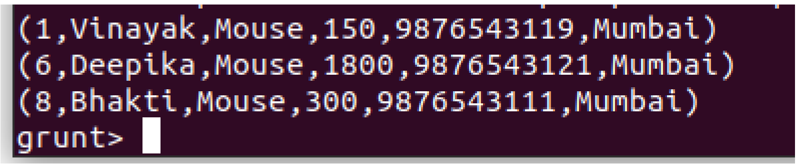


```
(5,Divya,Laptop,2500,9876543123,Banglore)
grunt>
```

8. **Display the details of customers from city ‘Mumbai’ who purchased ‘Mouse’.**



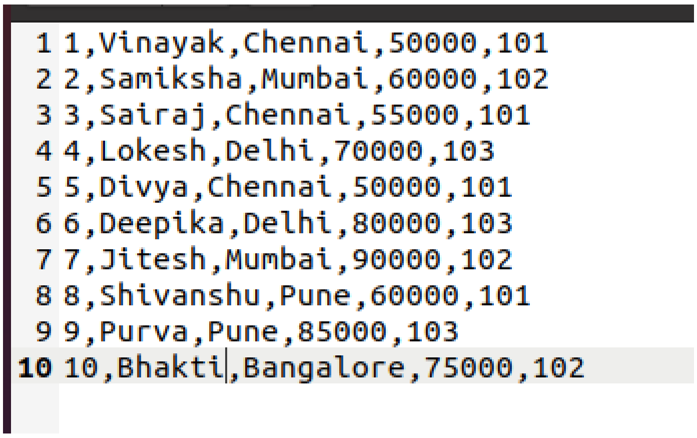
```
grunt> Mumbai_Mouse = FILTER Customer BY city == 'Mumbai' AND item_purchased ==
'Mouse';
grunt> dump Mumbai_Mouse;
```



```
(1,Vinayak,Mouse,150,9876543119,Mumbai)
(6,Deepika,Mouse,1800,9876543121,Mumbai)
(8,Bhakti,Mouse,300,9876543111,Mumbai)
grunt>
```

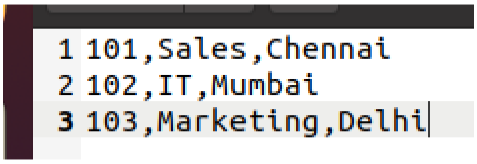
Q2) Perform the following in PIG

1. **Create emp.txt file with 6 records, file with following fields- eno, name, city, salary,did**



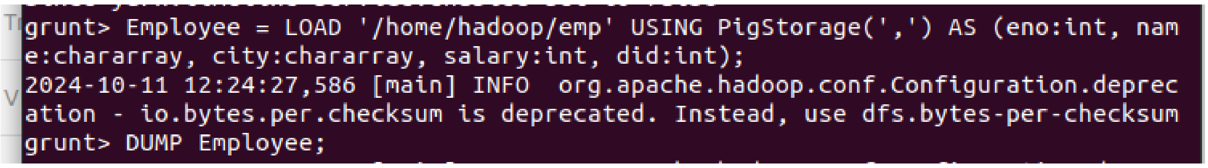
```
1,Vinayak,Chennai,50000,101
2,Samiksha,Mumbai,60000,102
3,Sairaj,Chennai,55000,101
4,Lokesh,Delhi,70000,103
5,Divya,Chennai,50000,101
6,Deepika,Delhi,80000,103
7,Jitesh,Mumbai,90000,102
8,Shivanshu,Pune,60000,101
9,Purva,Pune,85000,103
10,Bhakti,Bangalore,75000,102
```

2. **Create dept.txt file with three departments sales', 'IT','Marketing' with the fields- did, dname, location**

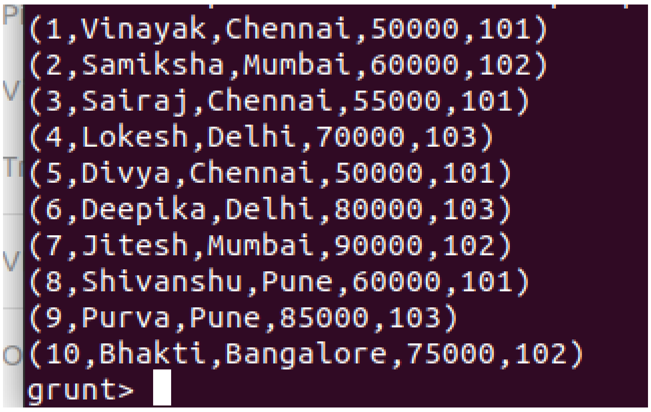


```
101,Sales,Chennai
102,IT,Mumbai
103,Marketing,Delhi
```

3. **. Create a relation Employee for the data given in emp .txt**

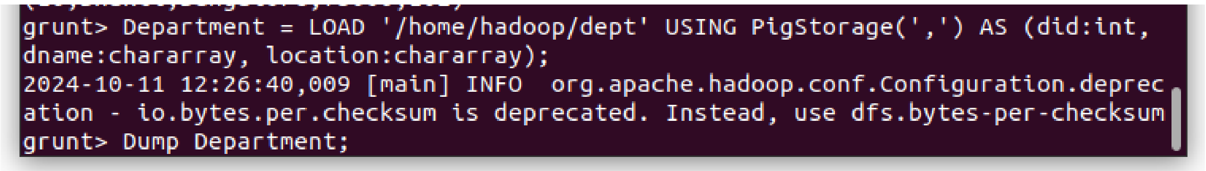


```
grunt> Employee = LOAD '/home/hadoop/emp' USING PigStorage(',') AS (eno:int, nam
e:chararray, city:chararray, salary:int, did:int);
2024-10-11 12:24:27,586 [main] INFO  org.apache.hadoop.conf.Configuration.deprec
ation - io.bytes.per.checksum is deprecated. Instead, use dfs.bytes-per-checksum
grunt> DUMP Employee;
```

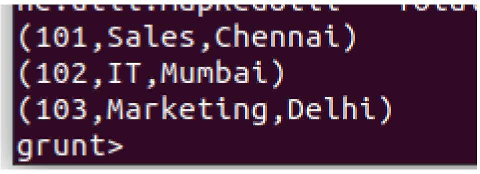


```
(1,Vinayak,Chennai,50000,101)
(2,Samiksha,Mumbai,60000,102)
(3,Sairaj,Chennai,55000,101)
(4,Lokesh,Delhi,70000,103)
(5,Divya,Chennai,50000,101)
(6,Deepika,Delhi,80000,103)
(7,Jitesh,Mumbai,90000,102)
(8,Shivanshu,Pune,60000,101)
(9,Purva,Pune,85000,103)
(10,Bhakti,Bangalore,75000,102)
grunt> 
```

4. **Create a relation Department and insert 5 records.**

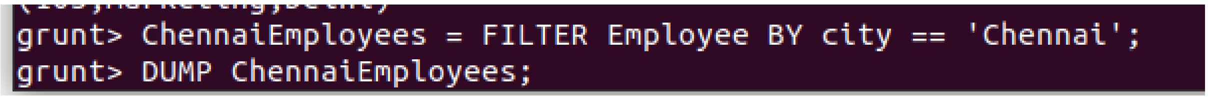


```
grunt> Department = LOAD '/home/hadoop/dept' USING PigStorage(',') AS (did:int,
dname:chararray, location:chararray);
2024-10-11 12:26:40,009 [main] INFO  org.apache.hadoop.conf.Configuration.deprec
ation - io.bytes.per.checksum is deprecated. Instead, use dfs.bytes-per-checksum
grunt> Dump Department;
```

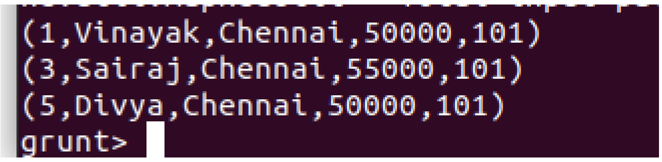


```
(101,Sales,Chennai)
(102,IT,Mumbai)
(103,Marketing,Delhi)
grunt>
```

5. **Display all the employees from city "Chennai"**

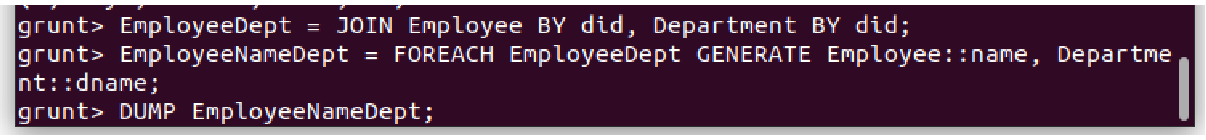


```
grunt> ChennaiEmployees = FILTER Employee BY city == 'Chennai';
grunt> DUMP ChennaiEmployees;
```

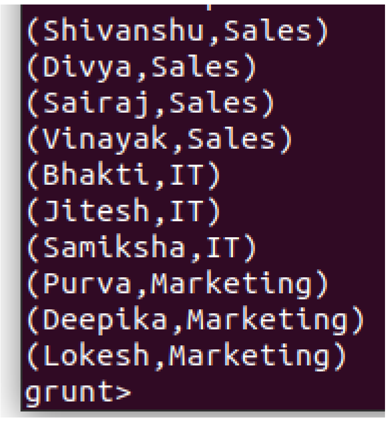


```
(1,Vinayak,Chennai,50000,101)
(3,Sairaj,Chennai,55000,101)
(5,Divya,Chennai,50000,101)
grunt>
```

6. **Display name of employees with their department name**

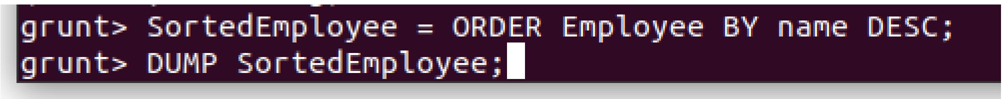


```
grunt> EmployeeDept = JOIN Employee BY did, Department BY did;
grunt> EmployeeNameDept = FOREACH EmployeeDept GENERATE Employee::name, Departme
nt::dname;
grunt> DUMP EmployeeNameDept;
```

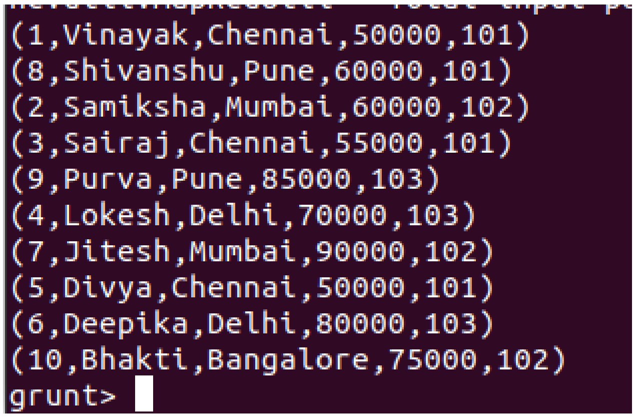


```
(Shivanshu,Sales)
(Divya,Sales)
(Sairaj,Sales)
(Vinayak,Sales)
(Bhakti,IT)
(Jitesh,IT)
(Samiksha,IT)
(Purva,Marketing)
(Deepika,Marketing)
(Lokesh,Marketing)
grunt>
```

7. **Sort the employee details according to their name in descending**

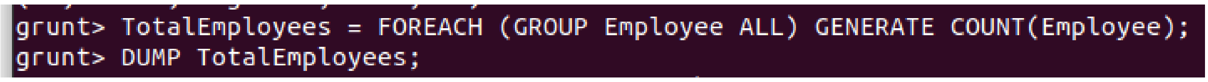


```
grunt> SortedEmployee = ORDER Employee BY name DESC;
grunt> DUMP SortedEmployee;
```

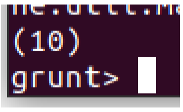


```
(1,Vinayak,Chennai,50000,101)
(8,Shivanshu,Pune,60000,101)
(2,Samiksha,Mumbai,60000,102)
(3,Sairaj,Chennai,55000,101)
(9,Purva,Pune,85000,103)
(4,Lokesh,Delhi,70000,103)
(7,Jitesh,Mumbai,90000,102)
(5,Divya,Chennai,50000,101)
(6,Deepika,Delhi,80000,103)
(10,Bhakti,Bangalore,75000,102)
grunt>
```

8. Display total number of employees.

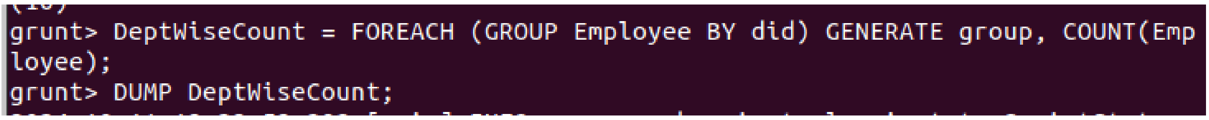


```
grunt> TotalEmployees = FOREACH (GROUP Employee ALL) GENERATE COUNT(Employee);
grunt> DUMP TotalEmployees;
```

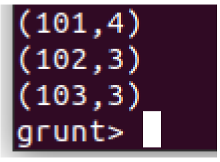


```
(10)
grunt>
```

9. Display the department wise employee count.

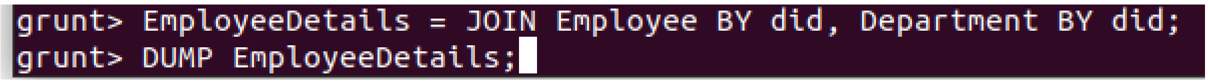


```
grunt> DeptWiseCount = FOREACH (GROUP Employee BY did) GENERATE group, COUNT(Emp
loyee);
grunt> DUMP DeptWiseCount;
```

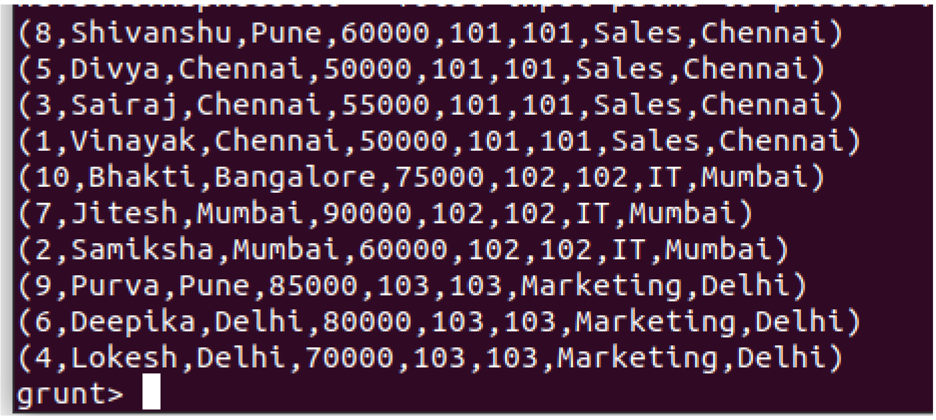


```
(101,4)
(102,3)
(103,3)
grunt>
```

10.Display employee and their department details

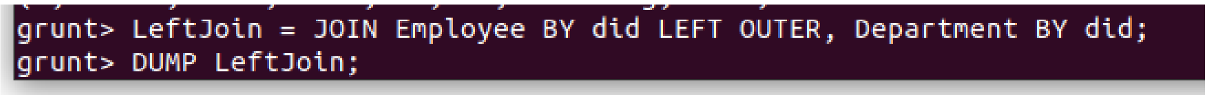


```
grunt> EmployeeDetails = JOIN Employee BY did, Department BY did;
grunt> DUMP EmployeeDetails;
```

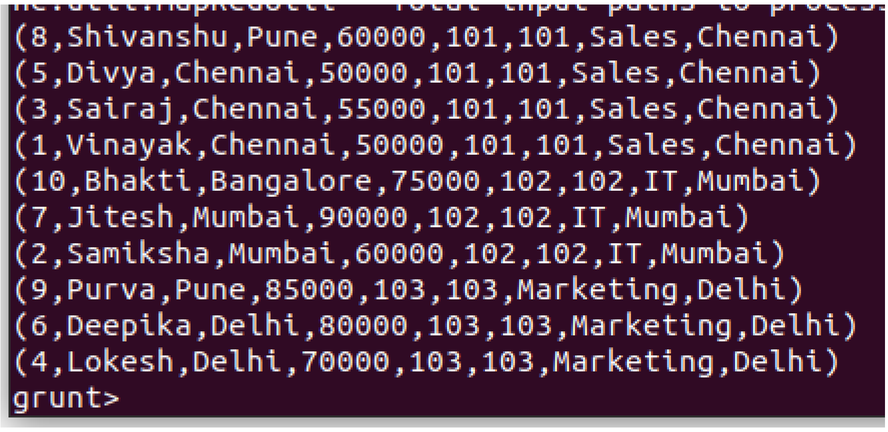


```
(8,Shivanshu,Pune,60000,101,101,Sales,Chennai)
(5,Divya,Chennai,50000,101,101,Sales,Chennai)
(3,Sairaj,Chennai,55000,101,101,Sales,Chennai)
(1,Vinayak,Chennai,50000,101,101,Sales,Chennai)
(10,Bhakti,Bangalore,75000,102,102,IT,Mumbai)
(7,Jitesh,Mumbai,90000,102,102,IT,Mumbai)
(2,Samiksha,Mumbai,60000,102,102,IT,Mumbai)
(9,Purva,Pune,85000,103,103,Marketing,Delhi)
(6,Deepika,Delhi,80000,103,103,Marketing,Delhi)
(4,Lokesh,Delhi,70000,103,103,Marketing,Delhi)
grunt>
```

11. **Perform Left Outer Join**



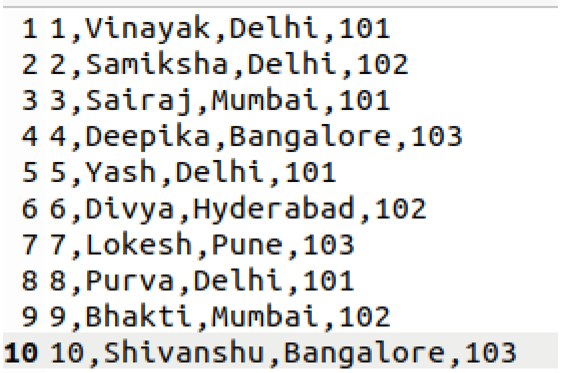
```
grunt> LeftJoin = JOIN Employee BY did LEFT OUTER, Department BY did;
grunt> DUMP LeftJoin;
```



```
(8,Shivanshu,Pune,60000,101,101,Sales,Chennai)
(5,Divya,Chennai,50000,101,101,Sales,Chennai)
(3,Sairaj,Chennai,55000,101,101,Sales,Chennai)
(1,Vinayak,Chennai,50000,101,101,Sales,Chennai)
(10,Bhakti,Bangalore,75000,102,102,IT,Mumbai)
(7,Jitesh,Mumbai,90000,102,102,IT,Mumbai)
(2,Samiksha,Mumbai,60000,102,102,IT,Mumbai)
(9,Purva,Pune,85000,103,103,Marketing,Delhi)
(6,Deepika,Delhi,80000,103,103,Marketing,Delhi)
(4,Lokesh,Delhi,70000,103,103,Marketing,Delhi)
grunt>
```

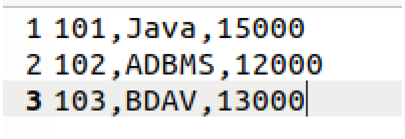
Q3) . Perform the following operations in PIG

1. **Create student.txt file with 10 records, file with following fields- Sid, sname, Saddress,cid**



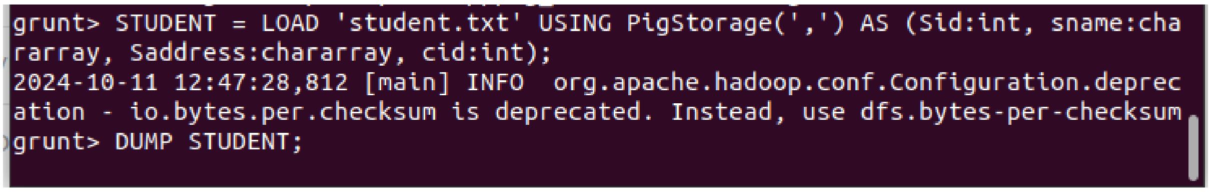
```
1,Vinayak,Delhi,101
2,Samiksha,Delhi,102
3,Sairaj,Mumbai,101
4,Deepika,Bangalore,103
5,Yash,Delhi,101
6,Divya,Hyderabad,102
7,Lokesh,Pune,103
8,Purva,Delhi,101
9,Bhakti,Mumbai,102
10,Shivanshu,Bangalore,103
```

2. **Create course.txt file for ‘Java’, ADBMS’ and ’BDAV’ courses, with the fields- cid ,cname,fees**

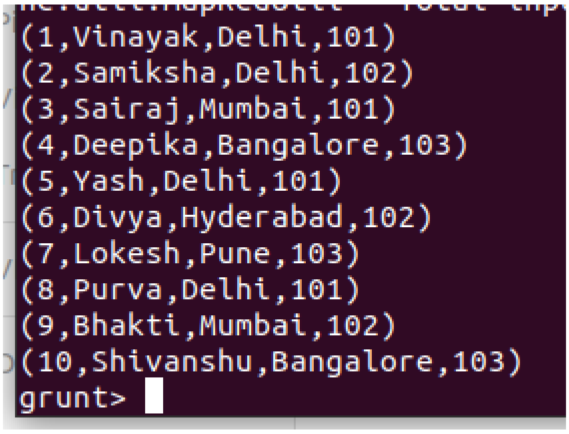


```
101,Java,15000
102,ADBMS,12000
103,BDAV,13000
```

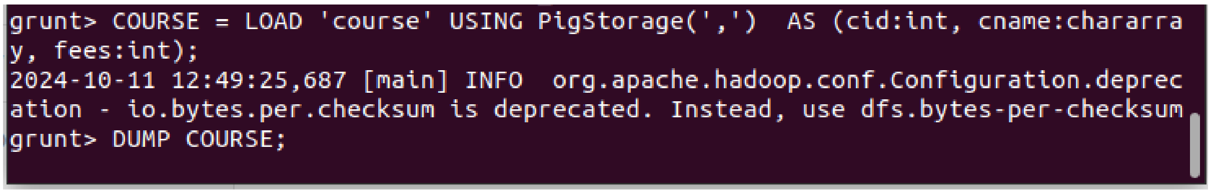
3. **Load the above file details into the relations STUDENT and COURSE.**



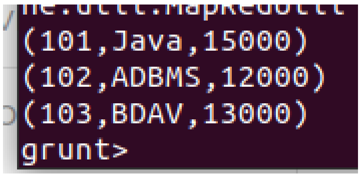
```
grunt> STUDENT = LOAD 'student.txt' USING PigStorage(',') AS (Sid:int, sname:cha
rarray, Saddress:chararray, cid:int);
2024-10-11 12:47:28,812 [main] INFO  org.apache.hadoop.conf.Configuration.deprec
ation - io.bytes.per.checksum is deprecated. Instead, use dfs.bytes-per-checksum
grunt> DUMP STUDENT;
```



```
(1,Vinayak,Delhi,101)
(2,Samiksha,Delhi,102)
(3,Sairaj,Mumbai,101)
(4,Deepika,Bangalore,103)
(5,Yash,Delhi,101)
(6,Divya,Hyderabad,102)
(7,Lokesh,Pune,103)
(8,Purva,Delhi,101)
(9,Bhakti,Mumbai,102)
(10,Shivanshu,Bangalore,103)
grunt>
```

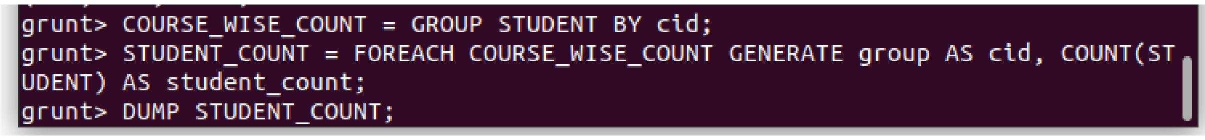


```
grunt> COURSE = LOAD 'course' USING PigStorage(',')  AS (cid:int, cname:chararra
y, fees:int);
2024-10-11 12:49:25,687 [main] INFO  org.apache.hadoop.conf.Configuration.deprec
ation - io.bytes.per.checksum is deprecated. Instead, use dfs.bytes-per-checksum
grunt> DUMP COURSE;
```

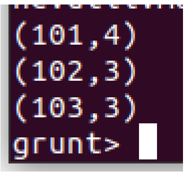


```
(101,Java,15000)
(102,ADBMS,12000)
(103,BDAV,13000)
grunt>
```

4. **Display course wise student count.**

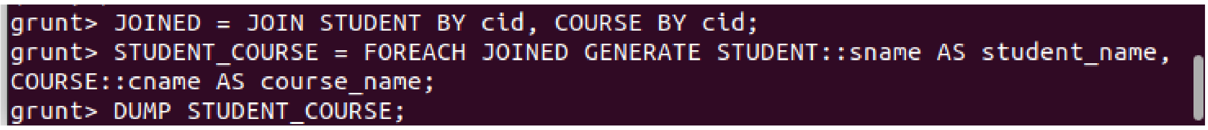


```
grunt> COURSE_WISE_COUNT = GROUP STUDENT BY cid;
grunt> STUDENT_COUNT = FOREACH COURSE_WISE_COUNT GENERATE group AS cid, COUNT(ST
UDENT) AS student_count;
grunt> DUMP STUDENT_COUNT;
```

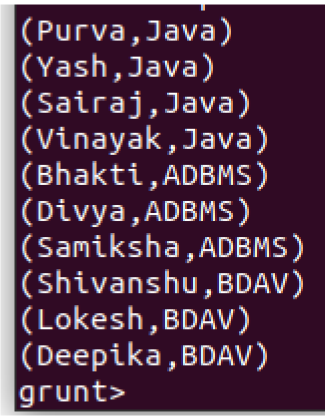


```
(101,4)
(102,3)
(103,3)
grunt>
```

5. **Display the student name and the course applied by each student**

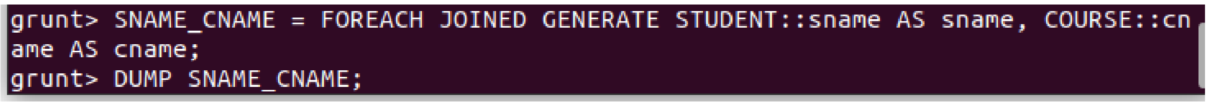


```
grunt> JOINED = JOIN STUDENT BY cid, COURSE BY cid;
grunt> STUDENT_COURSE = FOREACH JOINED GENERATE STUDENT::sname AS student_name,
COURSE::cname AS course_name;
grunt> DUMP STUDENT_COURSE;
```

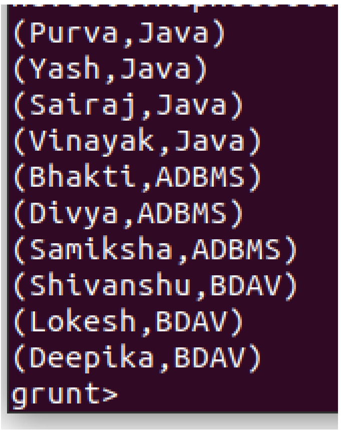


```
(Purva,Java)
(Yash,Java)
(Sairaj,Java)
(Vinayak,Java)
(Bhakti,ADBMS)
(Divya,ADBMS)
(Samiksha,ADBMS)
(Shivanshu,BDAV)
(Lokesh,BDAV)
(Deepika,BDAV)
grunt>
```

6. **Display sname and their cname.**



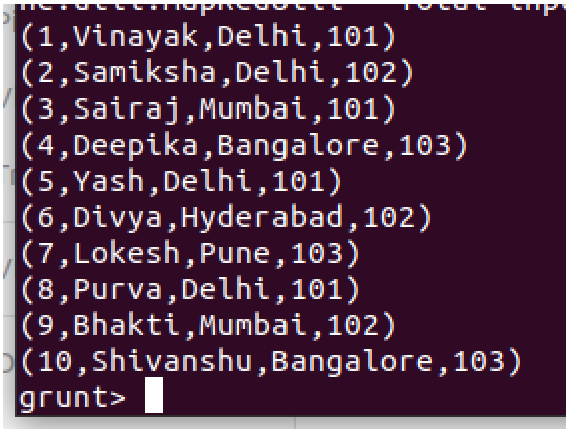
```
grunt> SNAME_CNAME = FOREACH JOINED GENERATE STUDENT::sname AS sname, COURSE::cn
ame AS cname;
grunt> DUMP SNAME_CNAME;
```



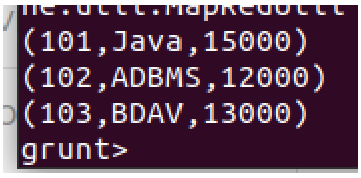
```
(Purva,Java)
(Yash,Java)
(Sairaj,Java)
(Vinayak,Java)
(Bhakti,ADBMS)
(Divya,ADBMS)
(Samiksha,ADBMS)
(Shivanshu,BDAV)
(Lokesh,BDAV)
(Deepika,BDAV)
grunt>
```

7. **Write a Pig script to perform the following operations:**

a. **Display the contents of STUDENT and COURSE relation**

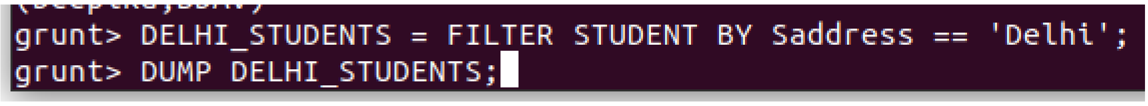


```
(1,Vinayak,Delhi,101)
(2,Samiksha,Delhi,102)
(3,Sairaj,Mumbai,101)
(4,Deepika,Bangalore,103)
(5,Yash,Delhi,101)
(6,Divya,Hyderabad,102)
(7,Lokesh,Pune,103)
(8,Purva,Delhi,101)
(9,Bhakti,Mumbai,102)
(10,Shivanshu,Bangalore,103)
grunt>
```

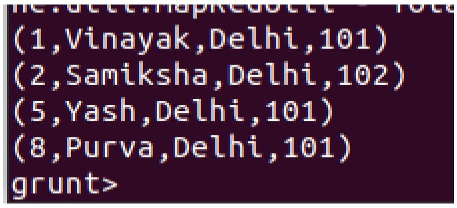


```
(101,Java,15000)
(102,ADBMS,12000)
(103,BDAV,13000)
grunt>
```

b. **Display the sid and sname who lives in "Delhi"**

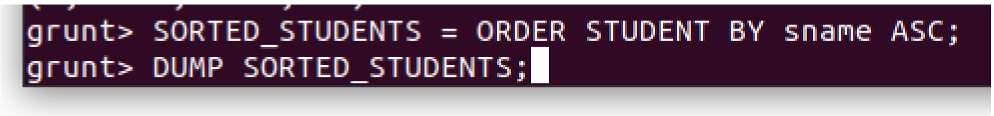


```
grunt> DELHI_STUDENTS = FILTER STUDENT BY Saddress == 'Delhi';
grunt> DUMP DELHI_STUDENTS;
```

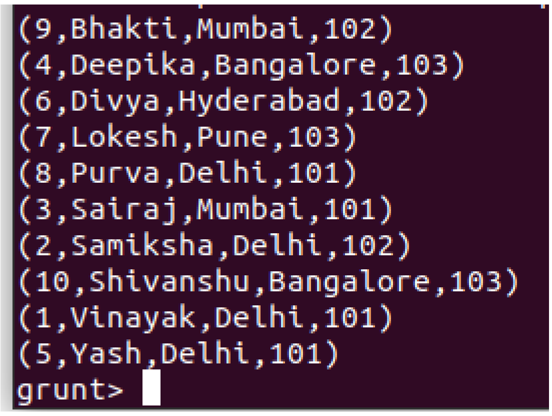


```
(1,Vinayak,Delhi,101)
(2,Samiksha,Delhi,102)
(5,Yash,Delhi,101)
(8,Purva,Delhi,101)
grunt>
```

c. **Display student details in ascending order of their name**



```
grunt> SORTED_STUDENTS = ORDER STUDENT BY sname ASC;
grunt> DUMP SORTED_STUDENTS;
```



```
(9,Bhakti,Mumbai,102)
(4,Deepika,Bangalore,103)
(6,Divya,Hyderabad,102)
(7,Lokesh,Pune,103)
(8,Purva,Delhi,101)
(3,Sairaj,Mumbai,101)
(2,Samiksha,Delhi,102)
(10,Shivanshu,Bangalore,103)
(1,Vinayak,Delhi,101)
(5,Yash,Delhi,101)
grunt>
```