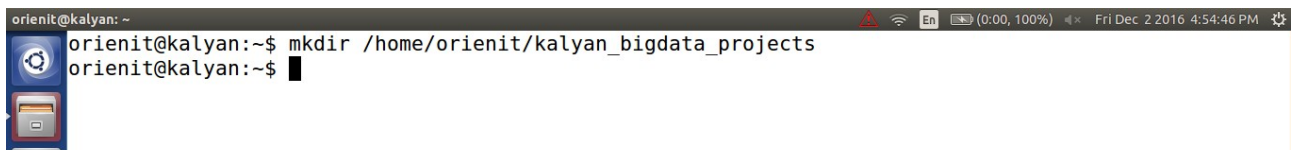


Kalyan Big Data Real Time Projects

How to generate **large amount of sample data** with simple techniques for **Big Data Projects**.
Follow the below commands to generate **large amount of sample data**.

i) Create '**kalyan_bigdata_projects**' folder in **user home** (i.e **/home/orienit**)

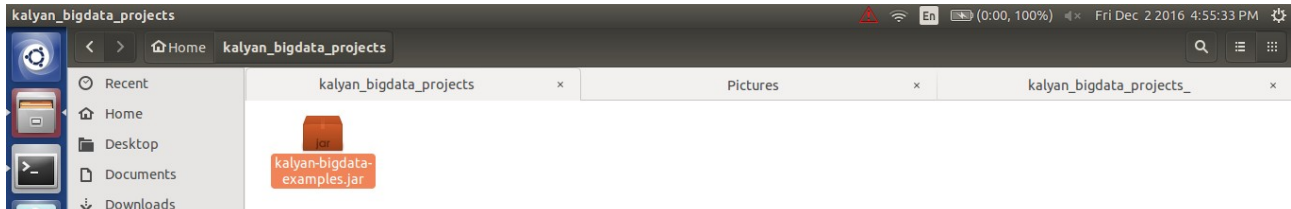
Command: `mkdir /home/orienit/kalyan_bigdata_projects`



ii) Download '**kalyan-bigdata-examples.jar**' jar file from this [link](https://github.com/kalyanhadooptraining/kalyan-bigdata-realtime-projects/blob/master/kalyan/kalyan-bigdata-examples.jar).

(<https://github.com/kalyanhadooptraining/kalyan-bigdata-realtime-projects/blob/master/kalyan/kalyan-bigdata-examples.jar>)

iii) Copy '**kalyan-bigdata-examples.jar**' jar file into '**/home/orienit/kalyan_bigdata_projects**' folder



We are going to learn below Use Cases

Use Case 1: Generating Sample **Server Logs** with simple command

Use Case 2: Generating Sample **Users** in **JSON** format with simple command

Use Case 3: Generating Sample **Users** in **CSV** format with simple command

Use Case 4: Generating Sample **Users** in **TSV** format with simple command

Use Case 5: Generating Sample **Users** in **DELIMITED** format with simple command

Use Case 6: Generating Sample **Product Log** in **JSON** format with simple command

Use Case 7: Generating Sample **Product Log** in **CSV** format with simple command

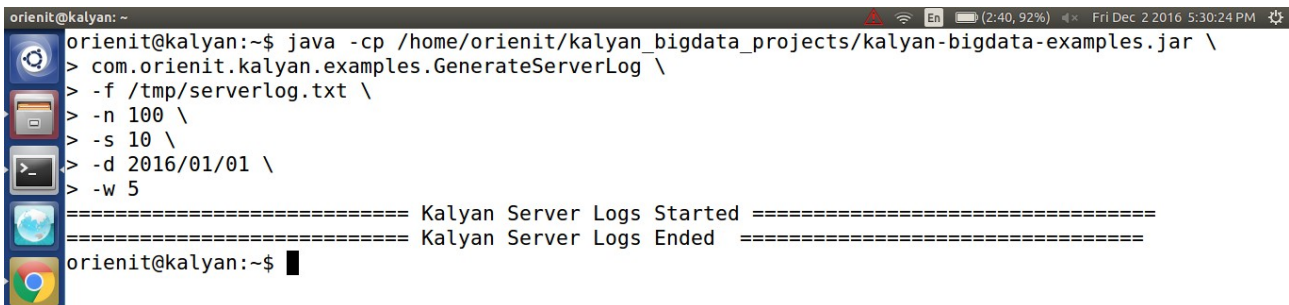
Use Case 8: Generating Sample **Product Log** in **TSV** format with simple command

Use Case 9: Generating Sample **Product Log** in **DELIMITED** format with simple command

Kalyan Big Data Projects – Use Case 1

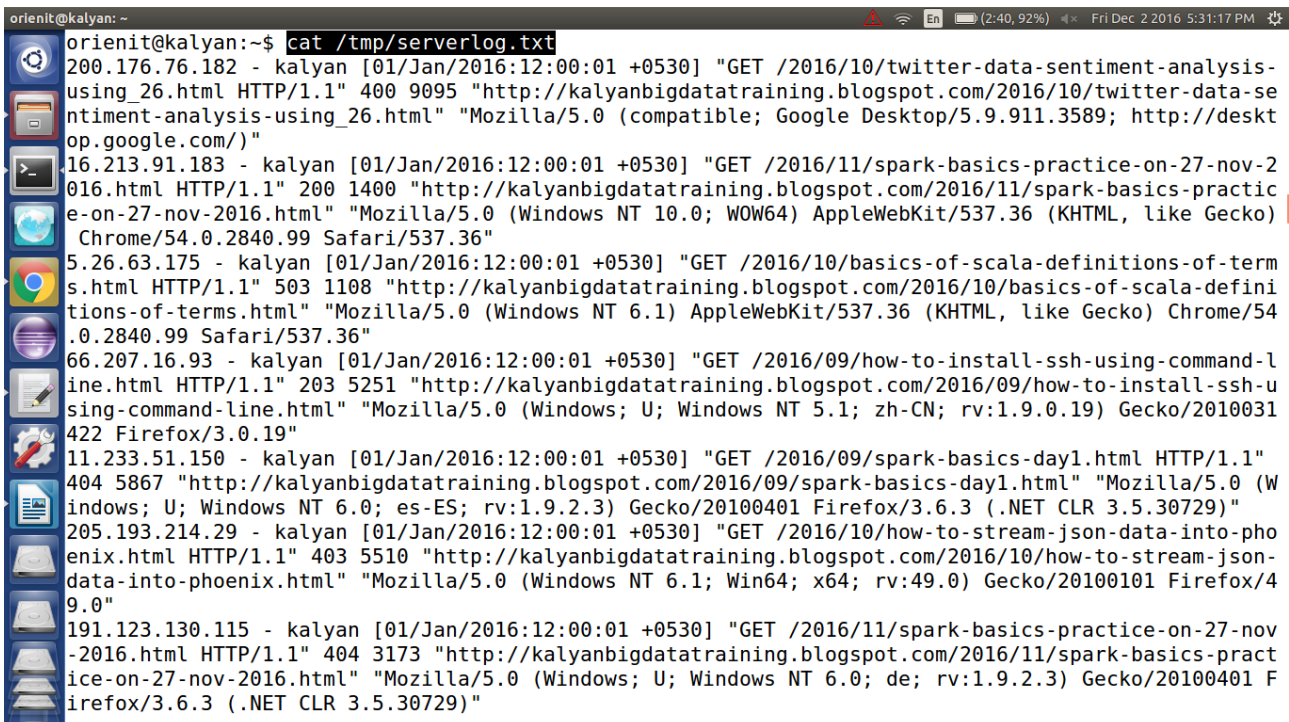
Generating Sample Server Logs with simple command

```
java -cp /home/orienit/kalyan_bigdata_projects/kalyan-bigdata-examples.jar \  
com.orienit.kalyan.examples.GenerateServerLog \  
-f /tmp/serverlog.txt \  
-n 100 \  
-s 10 \  
-d 2016/01/01 \  
-w 5
```



```
orienit@kalyan: ~  
orienit@kalyan:~$ java -cp /home/orienit/kalyan_bigdata_projects/kalyan-bigdata-examples.jar \  
> com.orienit.kalyan.examples.GenerateServerLog \  
> -f /tmp/serverlog.txt \  
> -n 100 \  
> -s 10 \  
> -d 2016/01/01 \  
> -w 5  
===== Kalyan Server Logs Started =====  
===== Kalyan Server Logs Ended =====  
orienit@kalyan:~$
```

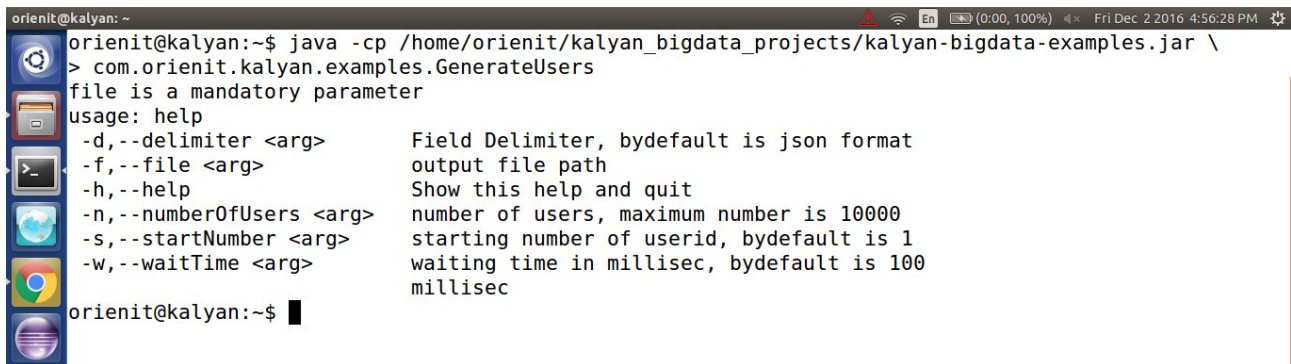
Read SERVER LOG data



```
orienit@kalyan: ~  
orienit@kalyan:~$ cat /tmp/serverlog.txt  
200.176.76.182 - kalyan [01/Jan/2016:12:00:01 +0530] "GET /2016/10/twitter-data-sentiment-analysis-using_26.html HTTP/1.1" 400 9095 "http://kalyanbigdatatraining.blogspot.com/2016/10/twitter-data-sentiment-analysis-using_26.html" "Mozilla/5.0 (compatible; Google Desktop/5.9.911.3589; http://desktop.google.com/)"  
16.213.91.183 - kalyan [01/Jan/2016:12:00:01 +0530] "GET /2016/11/spark-basics-practice-on-27-nov-2016.html HTTP/1.1" 200 1400 "http://kalyanbigdatatraining.blogspot.com/2016/11/spark-basics-practice-on-27-nov-2016.html" "Mozilla/5.0 (Windows NT 10.0; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/54.0.2840.99 Safari/537.36"  
5.26.63.175 - kalyan [01/Jan/2016:12:00:01 +0530] "GET /2016/10/basics-of-scala-definitions-of-terms.html HTTP/1.1" 503 1108 "http://kalyanbigdatatraining.blogspot.com/2016/10/basics-of-scala-definitions-of-terms.html" "Mozilla/5.0 (Windows NT 6.1) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/54.0.2840.99 Safari/537.36"  
66.207.16.93 - kalyan [01/Jan/2016:12:00:01 +0530] "GET /2016/09/how-to-install-ssh-using-command-line.html HTTP/1.1" 203 5251 "http://kalyanbigdatatraining.blogspot.com/2016/09/how-to-install-ssh-using-command-line.html" "Mozilla/5.0 (Windows; U; Windows NT 5.1; zh-CN; rv:1.9.0.19) Gecko/2010031422 Firefox/3.0.19"  
11.233.51.150 - kalyan [01/Jan/2016:12:00:01 +0530] "GET /2016/09/spark-basics-day1.html HTTP/1.1" 404 5867 "http://kalyanbigdatatraining.blogspot.com/2016/09/spark-basics-day1.html" "Mozilla/5.0 (Windows; U; Windows NT 6.0; es-ES; rv:1.9.2.3) Gecko/20100401 Firefox/3.6.3 (.NET CLR 3.5.30729)"  
205.193.214.29 - kalyan [01/Jan/2016:12:00:01 +0530] "GET /2016/10/how-to-stream-json-data-into-phoenix.html HTTP/1.1" 403 5510 "http://kalyanbigdatatraining.blogspot.com/2016/10/how-to-stream-json-data-into-phoenix.html" "Mozilla/5.0 (Windows NT 6.1; Win64; x64; rv:49.0) Gecko/20100101 Firefox/49.0"  
191.123.130.115 - kalyan [01/Jan/2016:12:00:01 +0530] "GET /2016/11/spark-basics-practice-on-27-nov-2016.html HTTP/1.1" 404 3173 "http://kalyanbigdatatraining.blogspot.com/2016/11/spark-basics-practice-on-27-nov-2016.html" "Mozilla/5.0 (Windows; U; Windows NT 6.0; de; rv:1.9.2.3) Gecko/20100401 Firefox/3.6.3 (.NET CLR 3.5.30729)"
```

Kalyan Big Data Projects – Use Case Generating Sample Users with simple command

```
java -cp /home/orienit/kalyan_bigdata_projects/kalyan-bigdata-examples.jar \  
com.orienit.kalyan.examples.GenerateUsers
```



```
orienit@kalyan: ~$ java -cp /home/orienit/kalyan_bigdata_projects/kalyan-bigdata-examples.jar \  
> com.orienit.kalyan.examples.GenerateUsers  
file is a mandatory parameter  
usage: help  
-d,--delimiter <arg>      Field Delimiter, bydefault is json format  
-f,--file <arg>           output file path  
-h,--help                 Show this help and quit  
-n,--numberOfUsers <arg> number of users, maximum number is 10000  
-s,--startNumber <arg>   starting number of userid, bydefault is 1  
-w,--waitTime <arg>      waiting time in millisec, bydefault is 100  
                           millisec  
orienit@kalyan:~$
```

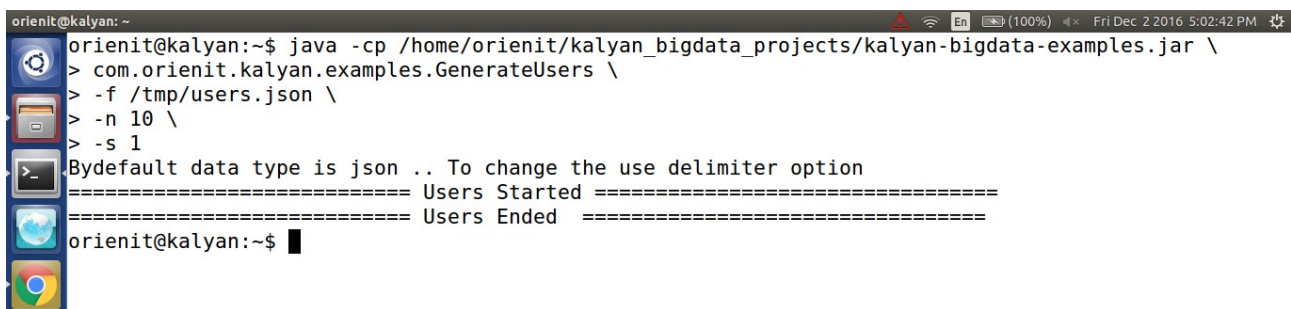
We can pass the different arguments for above command

- d => field delimiter like (tab, comma, semicolon, etc)
- f => output file path
- n => number of users, maximum number is 10000
- s => starting number of user id, bydefault is 1
- w => waiting time in milli sec, bydefault is 100 millisec

Kalyan Big Data Projects – Use Case 2

Generating Sample Users in JSON format with simple command

```
java -cp /home/orienit/kalyan_bigdata_projects/kalyan-bigdata-examples.jar \
com.orienit.kalyan.examples.GenerateUsers \
-f /tmp/users.json \
-n 10 \
-s 1
```



```
orienit@kalyan: ~$ java -cp /home/orienit/kalyan_bigdata_projects/kalyan-bigdata-examples.jar \
> com.orienit.kalyan.examples.GenerateUsers \
> -f /tmp/users.json \
> -n 10 \
> -s 1
By default data type is json .. To change the use delimiter option
===== Users Started =====
===== Users Ended =====
orienit@kalyan: ~$
```

Read JSON Data



```
orienit@kalyan: ~$ cat /tmp/users.json
{"userid":1,"username":"user1","password":"user1","email":"user1@gmail.com","country":"India","state":"Andhra Pradesh","city":"Guntur","dt":"2016-02-02 05:02:39"}
{"userid":2,"username":"user2","password":"user2","email":"user2@gmail.com","country":"US","state":"Washington","city":"Renton","dt":"2016-02-02 05:02:39"}
{"userid":3,"username":"user3","password":"user3","email":"user3@gmail.com","country":"US","state":"Hawaii","city":"Hanapepe","dt":"2016-02-02 05:02:39"}
{"userid":4,"username":"user4","password":"user4","email":"user4@gmail.com","country":"US","state":"Washington","city":"Bellingham","dt":"2016-02-02 05:02:39"}
{"userid":5,"username":"user5","password":"user5","email":"user5@gmail.com","country":"India","state":"Andhra Pradesh","city":"Kakinada","dt":"2016-02-02 05:02:39"}
{"userid":6,"username":"user6","password":"user6","email":"user6@gmail.com","country":"India","state":"Telangana","city":"Karimnagar","dt":"2016-02-02 05:02:39"}
{"userid":7,"username":"user7","password":"user7","email":"user7@gmail.com","country":"US","state":"New York","city":"Albany","dt":"2016-02-02 05:02:40"}
{"userid":8,"username":"user8","password":"user8","email":"user8@gmail.com","country":"India","state":"Karnataka","city":"Bidar","dt":"2016-02-02 05:02:40"}
{"userid":9,"username":"user9","password":"user9","email":"user9@gmail.com","country":"India","state":"Chennai","city":"Virugambakkam","dt":"2016-02-02 05:02:40"}
{"userid":10,"username":"user10","password":"user10","email":"user10@gmail.com","country":"US","state":"Hawaii","city":"Honolulu","dt":"2016-02-02 05:02:40"}
orienit@kalyan: ~$
```

Kalyan Big Data Projects – Use Case 3

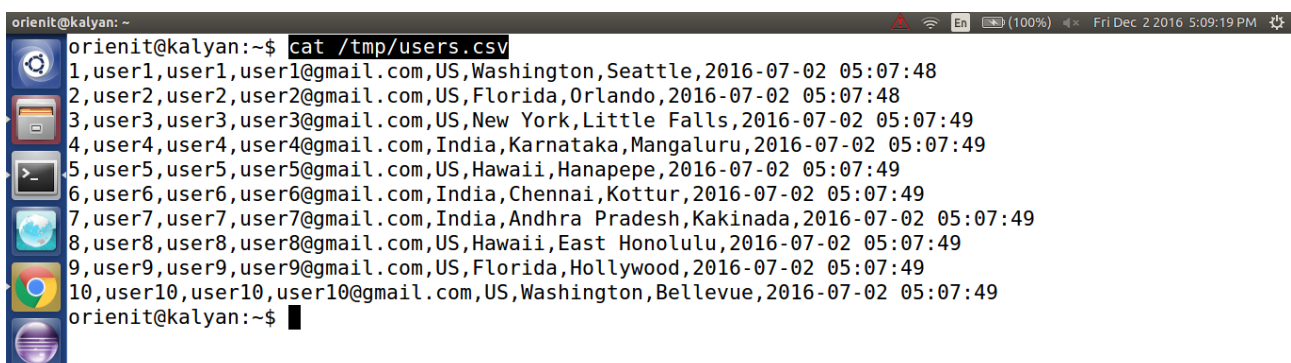
Generating Sample Users in CSV format with simple command

```
java -cp /home/orienit/kalyan_bigdata_projects/kalyan-bigdata-examples.jar \  
com.orienit.kalyan.examples.GenerateUsers \  
-f /tmp/users.csv \  
-d ',' \  
-n 10 \  
-s 1
```



```
orienit@kalyan: ~  
orienit@kalyan:~$ java -cp /home/orienit/kalyan_bigdata_projects/kalyan-bigdata-examples.jar \  
> com.orienit.kalyan.examples.GenerateUsers \  
> -f /tmp/users.csv \  
> -d ',' \  
> -n 10 \  
> -s 1  
===== Users Started =====  
===== Users Ended =====  
orienit@kalyan:~$
```

Read CSV data

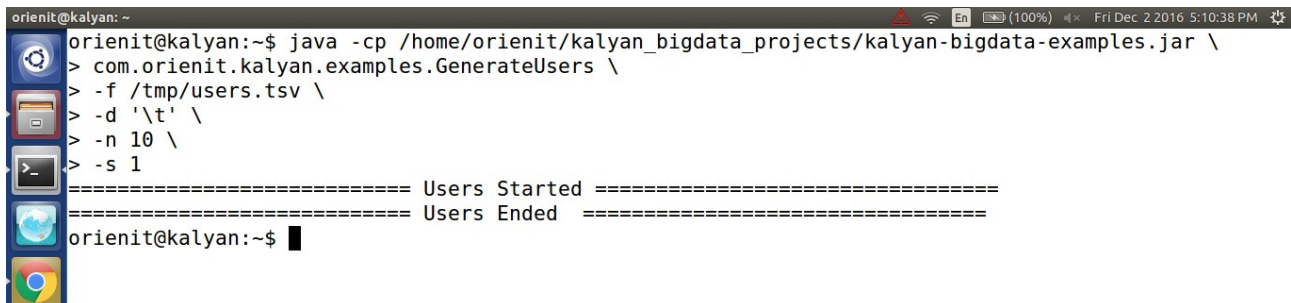


```
orienit@kalyan: ~  
orienit@kalyan:~$ cat /tmp/users.csv  
1,user1,user1,user1@gmail.com,US,Washington,Seattle,2016-07-02 05:07:48  
2,user2,user2,user2@gmail.com,US,Florida,Orlando,2016-07-02 05:07:48  
3,user3,user3,user3@gmail.com,US,New York,Little Falls,2016-07-02 05:07:49  
4,user4,user4,user4@gmail.com,India,Karnataka,Mangaluru,2016-07-02 05:07:49  
5,user5,user5,user5@gmail.com,US,Hawaii,Hanapepe,2016-07-02 05:07:49  
6,user6,user6,user6@gmail.com,India,Chennai,Kottur,2016-07-02 05:07:49  
7,user7,user7,user7@gmail.com,India,Andhra Pradesh,Kakinada,2016-07-02 05:07:49  
8,user8,user8,user8@gmail.com,US,Hawaii,East Honolulu,2016-07-02 05:07:49  
9,user9,user9,user9@gmail.com,US,Florida,Hollywood,2016-07-02 05:07:49  
10,user10,user10,user10@gmail.com,US,Washington,Bellevue,2016-07-02 05:07:49  
orienit@kalyan:~$
```


Kalyan Big Data Projects – Use Case 4

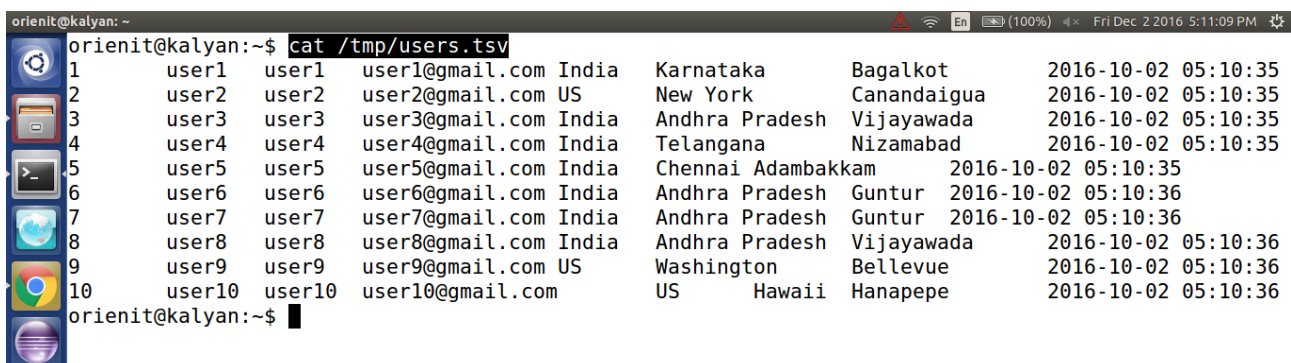
Generating Sample Users in TSV format with simple command

```
java -cp /home/orienit/kalyan_bigdata_projects/kalyan-bigdata-examples.jar \
com.orienit.kalyan.examples.GenerateUsers \
-f /tmp/users.tsv \
-d '\t' \
-n 10 \
-s 1
```



```
orienit@kalyan: ~$ java -cp /home/orienit/kalyan_bigdata_projects/kalyan-bigdata-examples.jar \
> com.orienit.kalyan.examples.GenerateUsers \
> -f /tmp/users.tsv \
> -d '\t' \
> -n 10 \
> -s 1
===== Users Started =====
===== Users Ended =====
orienit@kalyan:~$
```

Read TSV data



```
orienit@kalyan:~$ cat /tmp/users.tsv
1 user1 user1 user1@gmail.com India Karnataka Bagalkot 2016-10-02 05:10:35
2 user2 user2 user2@gmail.com US New York Canandaigua 2016-10-02 05:10:35
3 user3 user3 user3@gmail.com India Andhra Pradesh Vijayawada 2016-10-02 05:10:35
4 user4 user4 user4@gmail.com India Telangana Nizamabad 2016-10-02 05:10:35
5 user5 user5 user5@gmail.com India Chennai Adambakkam 2016-10-02 05:10:35
6 user6 user6 user6@gmail.com India Andhra Pradesh Guntur 2016-10-02 05:10:36
7 user7 user7 user7@gmail.com India Andhra Pradesh Guntur 2016-10-02 05:10:36
8 user8 user8 user8@gmail.com India Andhra Pradesh Vijayawada 2016-10-02 05:10:36
9 user9 user9 user9@gmail.com US Washington Bellevue 2016-10-02 05:10:36
10 user10 user10 user10@gmail.com US Hawaii Hanapepe 2016-10-02 05:10:36
orienit@kalyan:~$
```

Kalyan Big Data Projects – Use Case 5

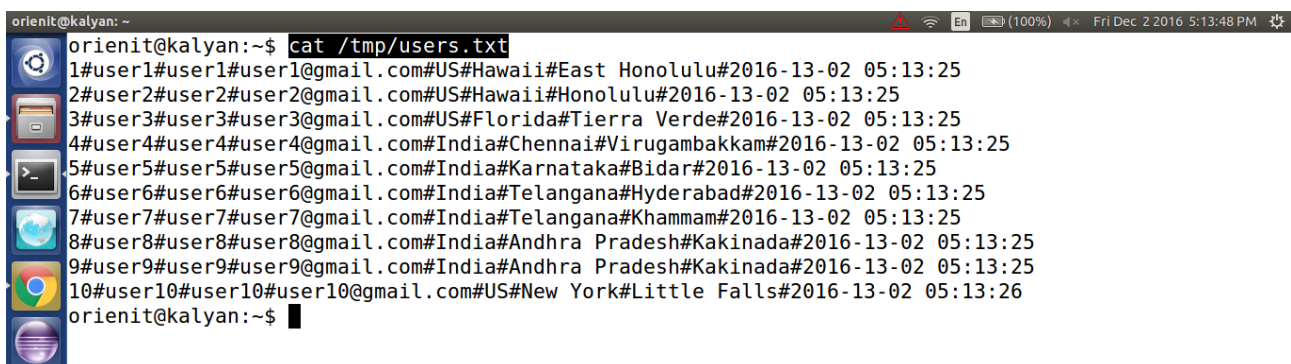
Generating Sample Users in DELIMITER format with simple command

```
java -cp /home/orienit/kalyan_bigdata_projects/kalyan-bigdata-examples.jar \  
com.orienit.kalyan.examples.GenerateUsers \  
-f /tmp/users.txt \  
-d '#' \  
-n 10 \  
-s 1
```



```
orienit@kalyan: ~  
orienit@kalyan:~$ java -cp /home/orienit/kalyan_bigdata_projects/kalyan-bigdata-examples.jar \  
> com.orienit.kalyan.examples.GenerateUsers \  
> -f /tmp/users.txt \  
> -d '#' \  
> -n 10 \  
> -s 1  
===== Users Started =====  
===== Users Ended =====  
orienit@kalyan:~$
```

Read Any **DELIMITED** Data

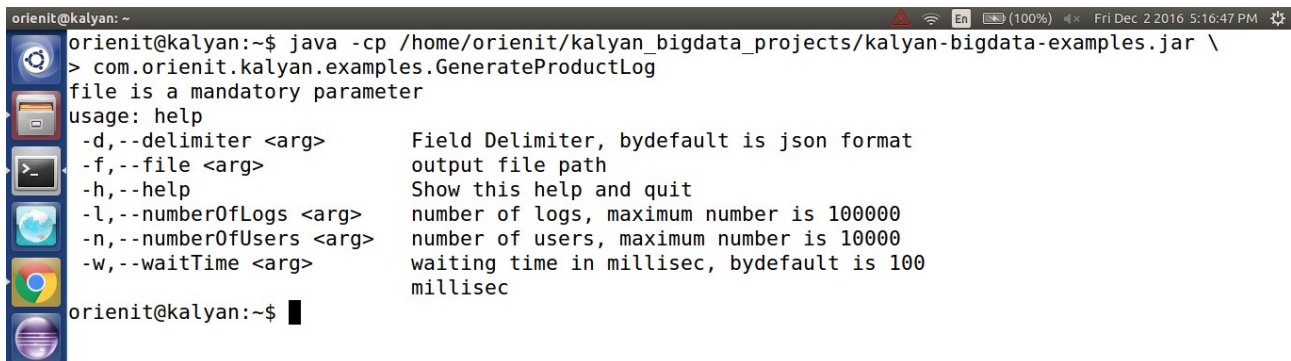


```
orienit@kalyan: ~  
orienit@kalyan:~$ cat /tmp/users.txt  
1#user1#user1#user1@gmail.com#US#Hawaii#East Honolulu#2016-13-02 05:13:25  
2#user2#user2#user2@gmail.com#US#Hawaii#Honolulu#2016-13-02 05:13:25  
3#user3#user3#user3@gmail.com#US#Florida#Tierra Verde#2016-13-02 05:13:25  
4#user4#user4#user4@gmail.com#India#Chennai#Virugambakkam#2016-13-02 05:13:25  
5#user5#user5#user5@gmail.com#India#Karnataka#Bidar#2016-13-02 05:13:25  
6#user6#user6#user6@gmail.com#India#Telangana#Hyderabad#2016-13-02 05:13:25  
7#user7#user7#user7@gmail.com#India#Telangana#Khammam#2016-13-02 05:13:25  
8#user8#user8#user8@gmail.com#India#Andhra Pradesh#Kakinada#2016-13-02 05:13:25  
9#user9#user9#user9@gmail.com#India#Andhra Pradesh#Kakinada#2016-13-02 05:13:25  
10#user10#user10#user10@gmail.com#US#New York#Little Falls#2016-13-02 05:13:26  
orienit@kalyan:~$
```

Kalyan Big Data Projects – Use Case

Generating Sample Product Log with simple command

```
java -cp /home/orienit/kalyan_bigdata_projects/kalyan-bigdata-examples.jar \  
com.orienit.kalyan.examples.GenerateProductLog
```



```
orienit@kalyan: ~  
orienit@kalyan:~$ java -cp /home/orienit/kalyan_bigdata_projects/kalyan-bigdata-examples.jar \  
> com.orienit.kalyan.examples.GenerateProductLog  
file is a mandatory parameter  
usage: help  
-d,--delimiter <arg>      Field Delimiter, bydefault is json format  
-f,--file <arg>          output file path  
-h,--help                Show this help and quit  
-l,--numberOfLogs <arg>  number of logs, maximum number is 100000  
-n,--numberOfUsers <arg> number of users, maximum number is 10000  
-w,--waitTime <arg>      waiting time in millisec, bydefault is 100  
                           millisec  
orienit@kalyan:~$
```

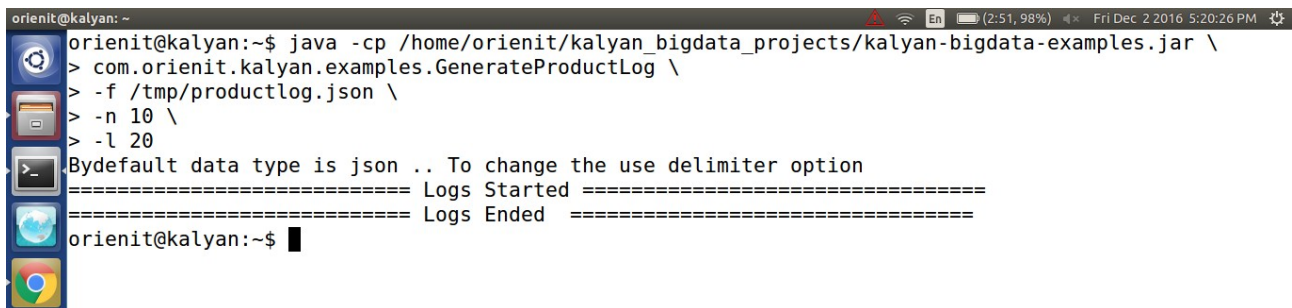
We can pass the different arguments for above command

- d => field delimiter like (tab, comma, semicolon, etc)
- f => output file path
- l => number of logs, maximum number is 100000
- n => number of users, maximum number is 10000
- w => waiting time in milli sec, bydefault is 100 millisec

Kalyan Big Data Projects – Use Case 6

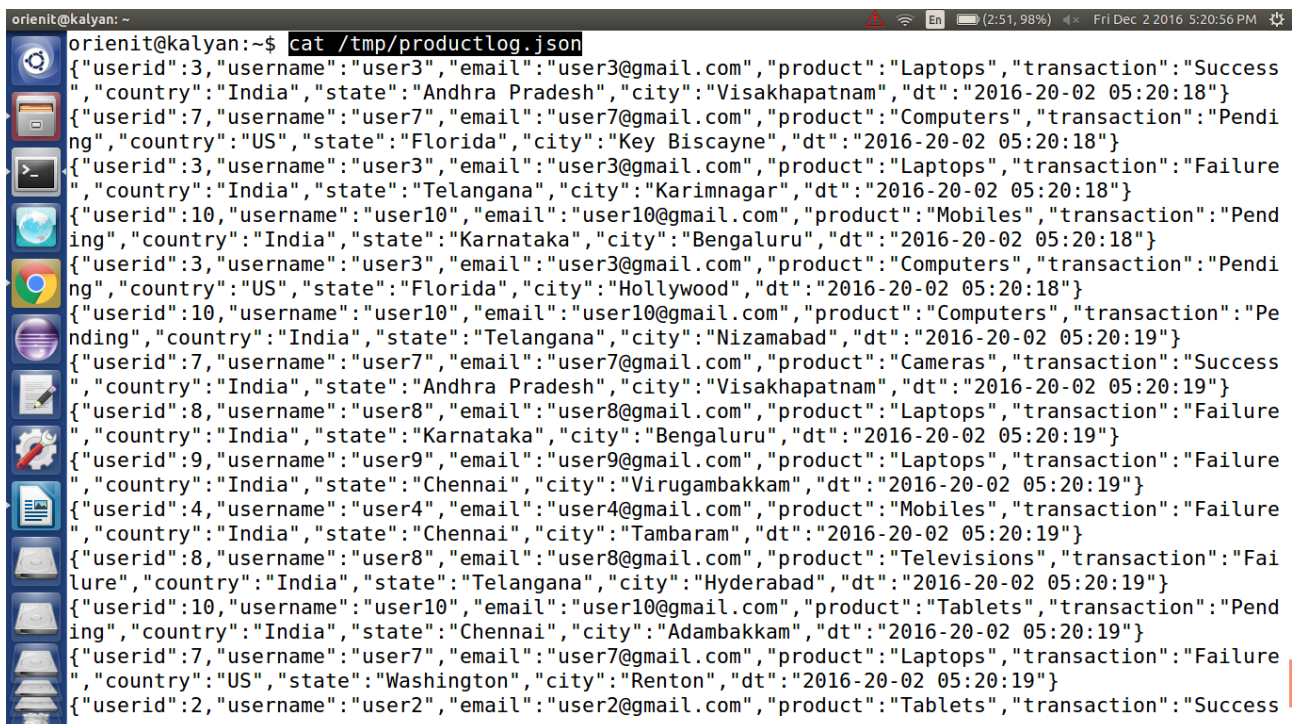
Generating Sample Product Log in JSON format with simple command

```
java -cp /home/orienit/kalyan_bigdata_projects/kalyan-bigdata-examples.jar \
com.orienit.kalyan.examples.GenerateProductLog \
-f /tmp/productlog.json \
-n 10 \
-l 20
```



```
orienit@kalyan: ~$ java -cp /home/orienit/kalyan_bigdata_projects/kalyan-bigdata-examples.jar \
> com.orienit.kalyan.examples.GenerateProductLog \
> -f /tmp/productlog.json \
> -n 10 \
> -l 20
By default data type is json .. To change the use delimiter option
===== Logs Started =====
===== Logs Ended =====
orienit@kalyan:~$
```

Read JSON data

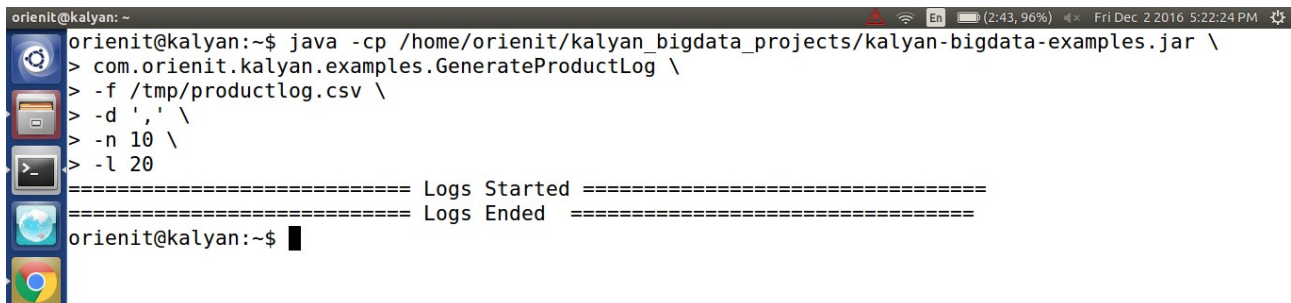


```
orienit@kalyan: ~$ cat /tmp/productlog.json
{"userid":3,"username":"user3","email":"user3@gmail.com","product":"Laptops","transaction":"Success",
"country":"India","state":"Andhra Pradesh","city":"Visakhapatnam","dt":"2016-20-02 05:20:18"}
{"userid":7,"username":"user7","email":"user7@gmail.com","product":"Computers","transaction":"Pending",
"country":"US","state":"Florida","city":"Key Biscayne","dt":"2016-20-02 05:20:18"}
{"userid":3,"username":"user3","email":"user3@gmail.com","product":"Laptops","transaction":"Failure",
"country":"India","state":"Telangana","city":"Karimnagar","dt":"2016-20-02 05:20:18"}
{"userid":10,"username":"user10","email":"user10@gmail.com","product":"Mobiles","transaction":"Pending",
"country":"India","state":"Karnataka","city":"Bengaluru","dt":"2016-20-02 05:20:18"}
{"userid":3,"username":"user3","email":"user3@gmail.com","product":"Computers","transaction":"Pending",
"country":"US","state":"Florida","city":"Hollywood","dt":"2016-20-02 05:20:18"}
{"userid":10,"username":"user10","email":"user10@gmail.com","product":"Computers","transaction":"Pending",
"country":"India","state":"Telangana","city":"Nizamabad","dt":"2016-20-02 05:20:19"}
{"userid":7,"username":"user7","email":"user7@gmail.com","product":"Cameras","transaction":"Success",
"country":"India","state":"Andhra Pradesh","city":"Visakhapatnam","dt":"2016-20-02 05:20:19"}
{"userid":8,"username":"user8","email":"user8@gmail.com","product":"Laptops","transaction":"Failure",
"country":"India","state":"Karnataka","city":"Bengaluru","dt":"2016-20-02 05:20:19"}
{"userid":9,"username":"user9","email":"user9@gmail.com","product":"Laptops","transaction":"Failure",
"country":"India","state":"Chennai","city":"Virugambakkam","dt":"2016-20-02 05:20:19"}
{"userid":4,"username":"user4","email":"user4@gmail.com","product":"Mobiles","transaction":"Failure",
"country":"India","state":"Chennai","city":"Tambaram","dt":"2016-20-02 05:20:19"}
{"userid":8,"username":"user8","email":"user8@gmail.com","product":"Televisions","transaction":"Failure",
"country":"India","state":"Telangana","city":"Hyderabad","dt":"2016-20-02 05:20:19"}
{"userid":10,"username":"user10","email":"user10@gmail.com","product":"Tablets","transaction":"Pending",
"country":"India","state":"Chennai","city":"Adambakkam","dt":"2016-20-02 05:20:19"}
{"userid":7,"username":"user7","email":"user7@gmail.com","product":"Laptops","transaction":"Failure",
"country":"US","state":"Washington","city":"Renton","dt":"2016-20-02 05:20:19"}
{"userid":2,"username":"user2","email":"user2@gmail.com","product":"Tablets","transaction":"Success"}
```

Kalyan Big Data Projects – Use Case 7

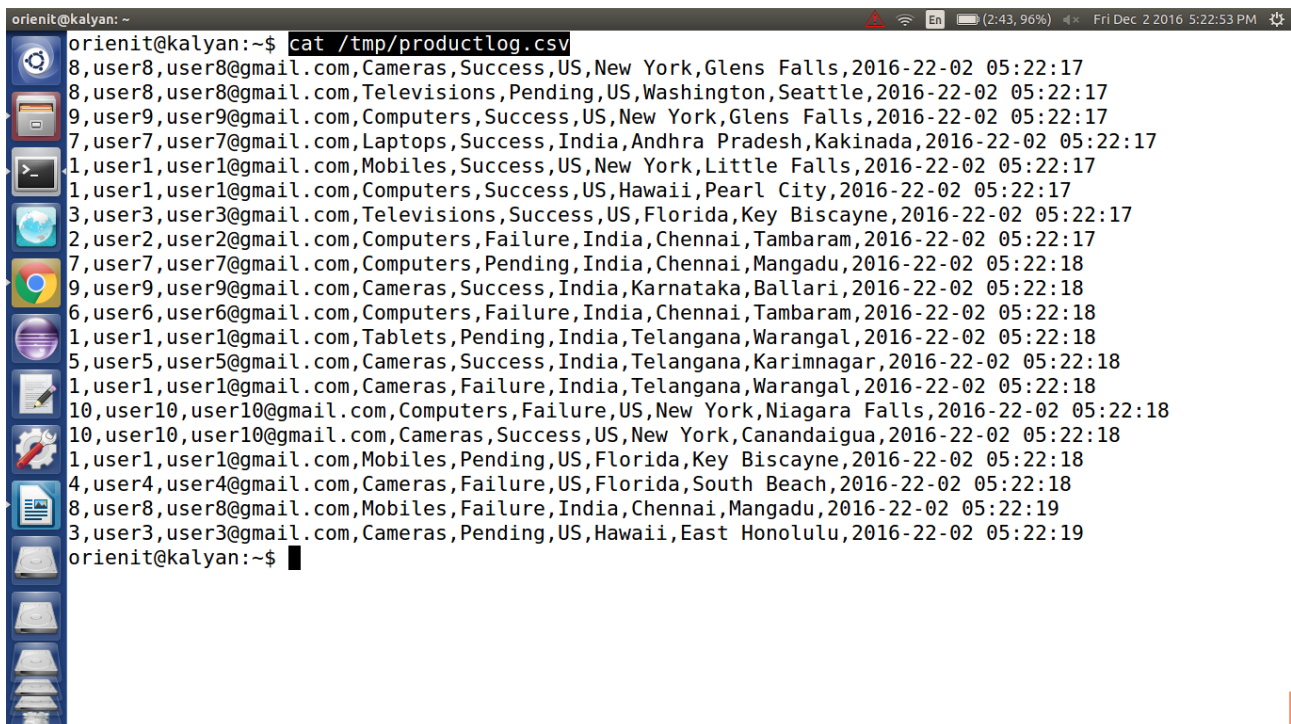
Generating Sample Product Log in CSV format with simple command

```
java -cp /home/orienit/kalyan_bigdata_projects/kalyan-bigdata-examples.jar \  
com.orienit.kalyan.examples.GenerateProductLog \  
-f /tmp/productlog.csv \  
-d ',' \  
-n 10 \  
-l 20
```



```
orienit@kalyan: ~$ java -cp /home/orienit/kalyan_bigdata_projects/kalyan-bigdata-examples.jar \  
> com.orienit.kalyan.examples.GenerateProductLog \  
> -f /tmp/productlog.csv \  
> -d ',' \  
> -n 10 \  
> -l 20  
===== Logs Started =====  
===== Logs Ended =====  
orienit@kalyan:~$
```

Read CSV data



```
orienit@kalyan:~$ cat /tmp/productlog.csv  
8,user8,user8@gmail.com,Cameras,Success,US,New York,Glens Falls,2016-22-02 05:22:17  
8,user8,user8@gmail.com,Televisions,Pending,US,Washington,Seattle,2016-22-02 05:22:17  
9,user9,user9@gmail.com,Computers,Success,US,New York,Glens Falls,2016-22-02 05:22:17  
7,user7,user7@gmail.com,Laptops,Success,India,Andhra Pradesh,Kakinada,2016-22-02 05:22:17  
1,user1,user1@gmail.com,Mobiles,Success,US,New York,Little Falls,2016-22-02 05:22:17  
1,user1,user1@gmail.com,Computers,Success,US,Hawaii,Pearl City,2016-22-02 05:22:17  
3,user3,user3@gmail.com,Televisions,Success,US,Florida,Key Biscayne,2016-22-02 05:22:17  
2,user2,user2@gmail.com,Computers,Failure,India,Chennai,Tambaram,2016-22-02 05:22:17  
7,user7,user7@gmail.com,Computers,Pending,India,Chennai,Mangadu,2016-22-02 05:22:18  
9,user9,user9@gmail.com,Cameras,Success,India,Karnataka,Ballari,2016-22-02 05:22:18  
6,user6,user6@gmail.com,Computers,Failure,India,Chennai,Tambaram,2016-22-02 05:22:18  
1,user1,user1@gmail.com,Tablets,Pending,India,Telangana,Warangal,2016-22-02 05:22:18  
5,user5,user5@gmail.com,Cameras,Success,India,Telangana,Karimnagar,2016-22-02 05:22:18  
1,user1,user1@gmail.com,Cameras,Failure,India,Telangana,Warangal,2016-22-02 05:22:18  
10,user10,user10@gmail.com,Computers,Failure,US,New York,Niagara Falls,2016-22-02 05:22:18  
10,user10,user10@gmail.com,Cameras,Success,US,New York,Canandaigua,2016-22-02 05:22:18  
1,user1,user1@gmail.com,Mobiles,Pending,US,Florida,Key Biscayne,2016-22-02 05:22:18  
4,user4,user4@gmail.com,Cameras,Failure,US,Florida,South Beach,2016-22-02 05:22:18  
8,user8,user8@gmail.com,Mobiles,Failure,India,Chennai,Mangadu,2016-22-02 05:22:19  
3,user3,user3@gmail.com,Cameras,Pending,US,Hawaii,East Honolulu,2016-22-02 05:22:19  
orienit@kalyan:~$
```

Kalyan Big Data Projects – Use Case 6

Generating Sample Product Log in TSV format with simple command

```
java -cp /home/orienit/kalyan_bigdata_projects/kalyan-bigdata-examples.jar \
com.orienit.kalyan.examples.GenerateProductLog \
-f /tmp/productlog.tsv \
-d '\t' \
-n 10 \
-l 20
```

```
orienit@kalyan: ~$ java -cp /home/orienit/kalyan_bigdata_projects/kalyan-bigdata-examples.jar \
> com.orienit.kalyan.examples.GenerateProductLog \
> -f /tmp/productlog.tsv \
> -d '\t' \
> -n 10 \
> -l 20
===== Logs Started =====
===== Logs Ended =====
orienit@kalyan: ~$
```

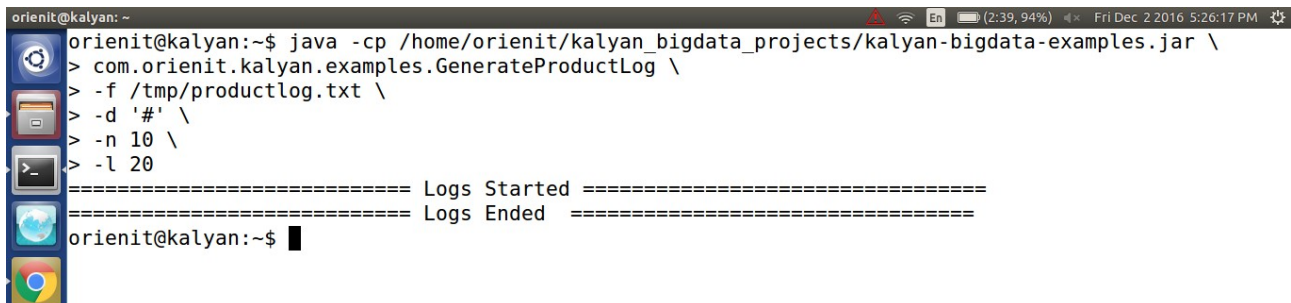
Read TSV data

```
orienit@kalyan: ~$ cat /tmp/productlog.tsv
10      user10   user10@gmail.com   Mobiles   Failure   India   Chennai   Madras   2016-23-02 05:23:51
1       user1    user1@gmail.com     Computers Failure   US       Hawaii    Mililani Town 2016-23-02 05:23:51
9       user9    user9@gmail.com     Laptops   Pending   US       New York   Little Falls 2016-23-02 05:23:51
6       user6    user6@gmail.com     Cameras   Success   India    Telangana  Nizamabad   2016-23-02 05:23:51
7       user7    user7@gmail.com     Laptops   Success   US       New York   Niagara Falls 2016-23-02 05:23:51
3       user3    user3@gmail.com     Mobiles   Pending   US       Hawaii    East Honolulu 2016-23-02 05:23:51
7       user7    user7@gmail.com     Cameras   Failure   US       Hawaii    Pupukea     2016-23-02 05:23:51
3       user3    user3@gmail.com     Televisions Pending   India    Karnataka  Bidar       2016-23-02 05:23:51
1       user1    user1@gmail.com     Mobiles   Pending   US       Washington Pasco       2016-23-02 05:23:52
6       user6    user6@gmail.com     Mobiles   Pending   India    Telangana  Karimnagar  2016-23-02 05:23:52
8       user8    user8@gmail.com     Televisions Success   India    Andhra Pradesh Rajahmundry 2016-23-02 05:23:52
3       user3    user3@gmail.com     Mobiles   Pending   India    Chennai   Madras     2016-23-02 05:23:52
3       user3    user3@gmail.com     Cameras   Failure   India    Chennai   Mangadu    2016-23-02 05:23:52
1       user1    user1@gmail.com     Computers Failure   India    Chennai   Mangadu    2016-23-02 05:23:52
1       user1    user1@gmail.com     Cameras   Success   US       New York   Glens Falls 2016-23-02 05:23:52
8       user8    user8@gmail.com     Tablets   Success   US       Florida   Indian Creek 2016-23-02 05:23:52
3       user3    user3@gmail.com     Computers Failure   US       Hawaii    Honolulu    2016-23-02 05:23:52
7       user7    user7@gmail.com     Televisions Failure   US       New York   Albany      2016-23-02 05:23:52
```


Kalyan Big Data Projects – Use Case 9

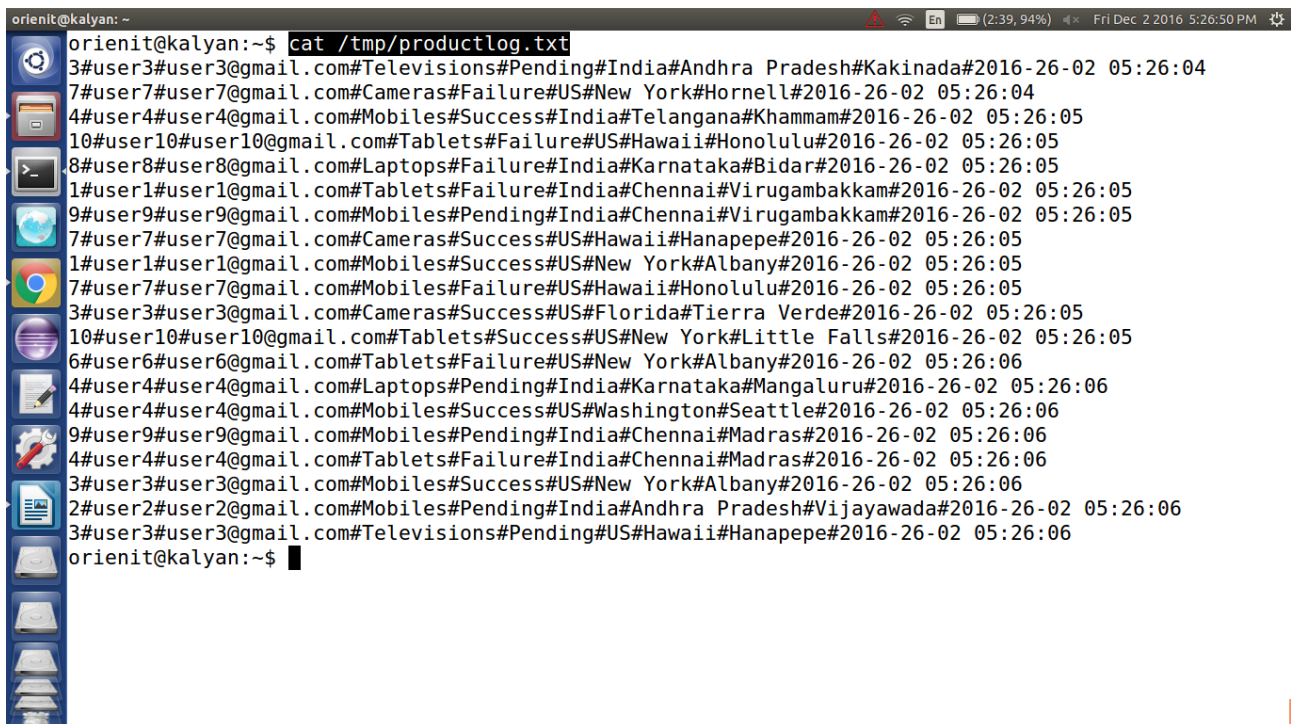
Generating Sample Product Log in DELIMITED format with simple command

```
java -cp /home/orienit/kalyan_bigdata_projects/kalyan-bigdata-examples.jar \  
com.orienit.kalyan.examples.GenerateProductLog \  
-f /tmp/productlog.txt \  
-d '#' \  
-n 10 \  
-l 20
```



```
orienit@kalyan: ~$ java -cp /home/orienit/kalyan_bigdata_projects/kalyan-bigdata-examples.jar \  
> com.orienit.kalyan.examples.GenerateProductLog \  
> -f /tmp/productlog.txt \  
> -d '#' \  
> -n 10 \  
> -l 20  
===== Logs Started =====  
===== Logs Ended =====  
orienit@kalyan:~$
```

Read Any **DELIMITED** data



```
orienit@kalyan: ~$ cat /tmp/productlog.txt  
3#user3#user3@gmail.com#Televisions#Pending#India#Andhra Pradesh#Kakinada#2016-26-02 05:26:04  
7#user7#user7@gmail.com#Cameras#Failure#US#New York#Hornell#2016-26-02 05:26:04  
4#user4#user4@gmail.com#Mobiles#Success#India#Telangana#Khammam#2016-26-02 05:26:05  
10#user10#user10@gmail.com#Tablets#Failure#US#Hawaii#Honolulu#2016-26-02 05:26:05  
8#user8#user8@gmail.com#Laptops#Failure#India#Karnataka#Bidar#2016-26-02 05:26:05  
1#user1#user1@gmail.com#Tablets#Failure#India#Chennai#Virugambakkam#2016-26-02 05:26:05  
9#user9#user9@gmail.com#Mobiles#Pending#India#Chennai#Virugambakkam#2016-26-02 05:26:05  
7#user7#user7@gmail.com#Cameras#Success#US#Hawaii#Hanapepe#2016-26-02 05:26:05  
1#user1#user1@gmail.com#Mobiles#Success#US#New York#Albany#2016-26-02 05:26:05  
7#user7#user7@gmail.com#Mobiles#Failure#US#Hawaii#Honolulu#2016-26-02 05:26:05  
3#user3#user3@gmail.com#Cameras#Success#US#Florida#Tierra Verde#2016-26-02 05:26:05  
10#user10#user10@gmail.com#Tablets#Success#US#New York#Little Falls#2016-26-02 05:26:05  
6#user6#user6@gmail.com#Tablets#Failure#US#New York#Albany#2016-26-02 05:26:06  
4#user4#user4@gmail.com#Laptops#Pending#India#Karnataka#Mangaluru#2016-26-02 05:26:06  
4#user4#user4@gmail.com#Mobiles#Success#US#Washington#Seattle#2016-26-02 05:26:06  
9#user9#user9@gmail.com#Mobiles#Pending#India#Chennai#Madras#2016-26-02 05:26:06  
4#user4#user4@gmail.com#Tablets#Failure#India#Chennai#Madras#2016-26-02 05:26:06  
3#user3#user3@gmail.com#Mobiles#Success#US#New York#Albany#2016-26-02 05:26:06  
2#user2#user2@gmail.com#Mobiles#Pending#India#Andhra Pradesh#Vijayawada#2016-26-02 05:26:06  
3#user3#user3@gmail.com#Televisions#Pending#US#Hawaii#Hanapepe#2016-26-02 05:26:06  
orienit@kalyan:~$
```