APPLIED BIG DATA ANALYTICS THEORY

SUBMITTED BY

SHRAVYA RAMESH ERABATHINI

2023-2024

HSNC UNIVERSITY

MASTERS OF SCIENCE IN
INFORMATION TECHNOLOGY
KISHINCHAND CHELLARAM COLLEGE
D.W.ROAD, CHURCHGATE, MUMBAI–400 020.

Signature

SUBJECT CODE - MS-SIT-308

APPLIED BIG DATA ANALYTICS THEORY

Signature

KISHINCHAND CHELLARAM COLLEGE
CHURCHGATE, MUMBAI – 400 020.

DEPARTMENT OF INFORMATION TECHNOLOGY

M.SC. PART- II

CERTIFICATE

This is to certify that the practical done at **K.C. College** by

MR/MS. ____

(Seat No: ____________________) in partial fulfillment for M.SC. (I.T.) Degree Examination has been found satisfactory. This Practical journal had not been submitted for any other examination and does not form part of any other course undergone by the candidate.

Signature	**Signature**	**Signature**
Lecturer-In-Charge	**External Examiner**	**Course Coordination**
Guided By	**Examined By**	**Certified By**

College Stamp

Signature

INDEX

Signature

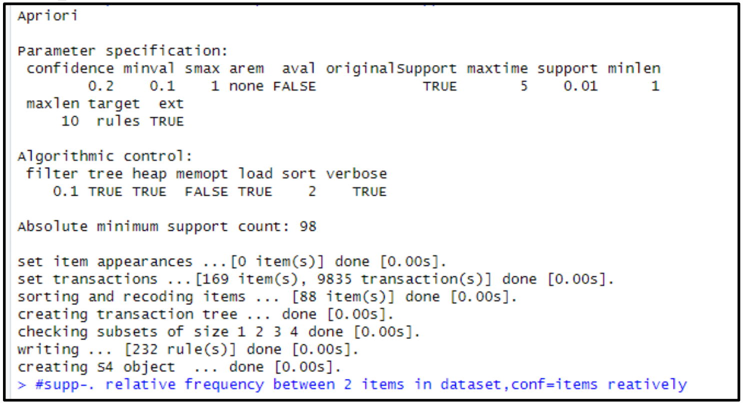
Practical 01

Aim: Apriori Algorithm

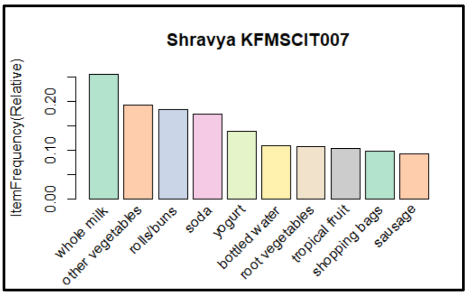
Code:

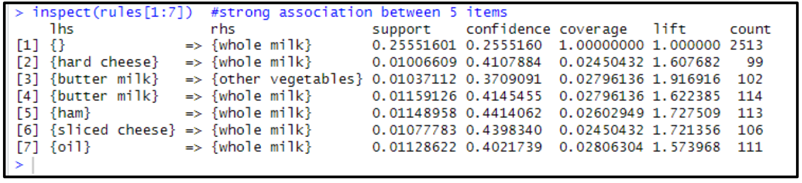
```
install.packages("arules")  #analysis package used in transactional dataset
install.packages("arulesViz") #Vizualization
install.packages("RColorBrewer") #Coloring the graph

#Loading Libraries
library(arules)
library(arulesViz)
library(RColorBrewer)

#import
data("Groceries")
rules<-apriori(Groceries,parameter = list(supp=0.01,conf=0.2))
#supp-. relative frequency between 2 items in dataset,conf=items reatively

#using inspect function
inspect(rules[1:7])  #strong association between 7 items
#Using itemFrequencyPlot() function
arules::itemFrequencyPlot(Groceries,topN=10,
                    col=brewer.pal(8,'Pastel2'),
                    main="Shravya KFMSCIT007",
                    type="relative",
                    ylab="ItemFrequency(Relative)")
```

Output:



```
Apriori

Parameter specification:
 confidence minval smax arem  aval originalSupport maxtime support minlen
        0.2    0.1    1 none FALSE            TRUE       5    0.01      1
 maxlen target  ext
     10  rules TRUE

Algorithmic control:
 filter tree heap memopt load sort verbose
    0.1 TRUE TRUE  FALSE TRUE    2    TRUE

Absolute minimum support count: 98

set item appearances ...[0 item(s)] done [0.00s].
set transactions ...[169 item(s), 9835 transaction(s)] done [0.00s].
sorting and recoding items ... [88 item(s)] done [0.00s].
creating transaction tree ... done [0.00s].
checking subsets of size 1 2 3 4 done [0.00s].
writing ... [232 rule(s)] done [0.00s].
creating S4 object  ... done [0.00s].
> #supp-. relative frequency between 2 items in dataset,conf=items reatively
```





```
> inspect(rules[1:7])  #strong association between 5 items
    lhs                 rhs                support    confidence coverage   lift     count
[1] {}               => {whole milk}       0.25551601 0.2555160  1.00000000 1.000000 2513
[2] {hard cheese}    => {whole milk}       0.01006609 0.4107884  0.02450432 1.607682   99
[3] {butter milk}    => {other vegetables} 0.01037112 0.3709091  0.02796136 1.916916  102
[4] {butter milk}    => {whole milk}       0.01159126 0.4145455  0.02796136 1.622385  114
[5] {ham}            => {whole milk}       0.01148958 0.4414062  0.02602949 1.727509  113
[6] {sliced cheese} => {whole milk}       0.01077783 0.4398340  0.02450432 1.721356  106
[7] {oil}            => {whole milk}       0.01128622 0.4021739  0.02806304 1.573968  111
> 
```

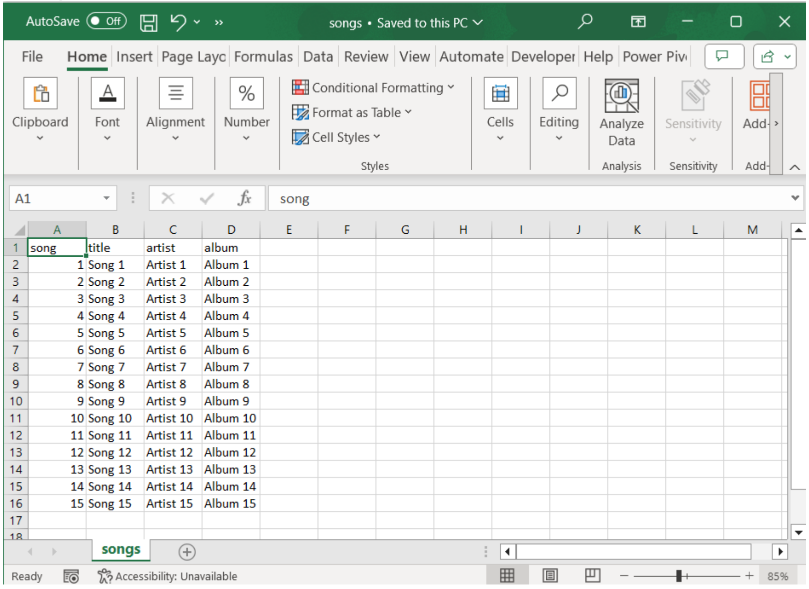
Signature

NAME: SHRAVYA ERABATHINI ROLL NO.:07

Practical 02

Aim: Write a python program to pick the content for billboards from the big data

Songs.csv



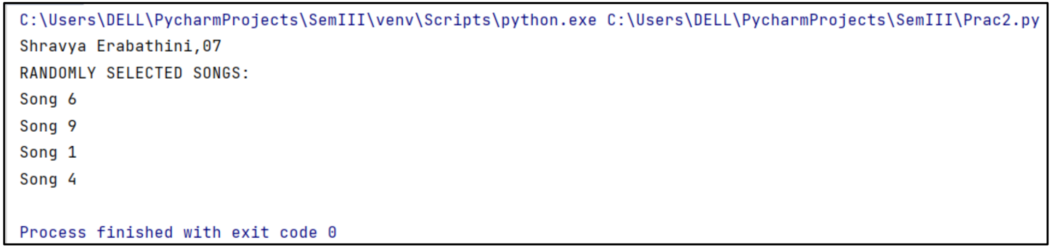
song	title	artist	album
1	Song 1	Artist 1	Album 1
2	Song 2	Artist 2	Album 2
3	Song 3	Artist 3	Album 3
4	Song 4	Artist 4	Album 4
5	Song 5	Artist 5	Album 5
6	Song 6	Artist 6	Album 6
7	Song 7	Artist 7	Album 7
8	Song 8	Artist 8	Album 8
9	Song 9	Artist 9	Album 9
10	Song 10	Artist 10	Album 10
11	Song 11	Artist 11	Album 11
12	Song 12	Artist 12	Album 12
13	Song 13	Artist 13	Album 13
14	Song 14	Artist 14	Album 14
15	Song 15	Artist 15	Album 15

(A) Using Python with CSV

Code:

```
print("Shravya Erabathini,07")
import csv
import random

def get_random_songs(csv_file, num_songs):
    with open(csv_file, 'r') as file:
        reader = csv.DictReader(file)
        all_songs = list(reader)
    random.shuffle(all_songs)
    random_songs = all_songs[:num_songs]
    return random_songs

csv_file = 'songs.csv'
num_songs = 4

random_songs = get_random_songs(csv_file, num_songs)

print("RANDOMLY SELECTED SONGS:")
for song in random_songs:
    print(f"{song['title']}")
```

Output:



```
C:\Users\DELL\PycharmProjects\SemIII\venv\Scripts\python.exe C:\Users\DELL\PycharmProjects\SemIII\Prac2.py
Shravya Erabathini,07
RANDOMLY SELECTED SONGS:
Song 6
Song 9
Song 1
Song 4

Process finished with exit code 0
```

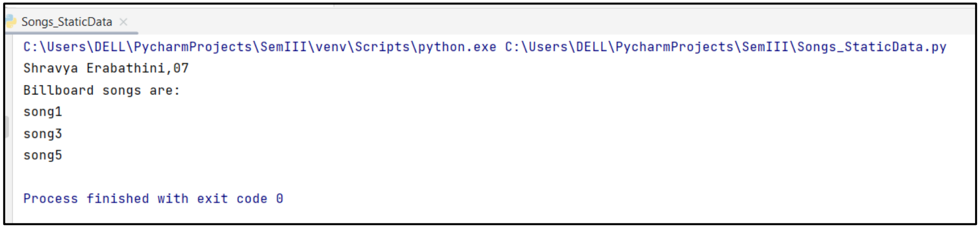
Signature

(B) Using Python with Static Data

Code:

```
print("Shravya Erabathini,07")
import random
def billboard_songs(songs,num_songs):
    random.shuffle(songs)
    bill_songs=songs[:num_songs]
    return bill_songs
all_songs=["song1","song2","song3","song4","song5"]
num_bill_song=3
bill_songs=billboard_songs(all_songs,num_bill_song)
print("Billboard songs are: ")
for song in bill_songs:
    print(song)
```

Output:



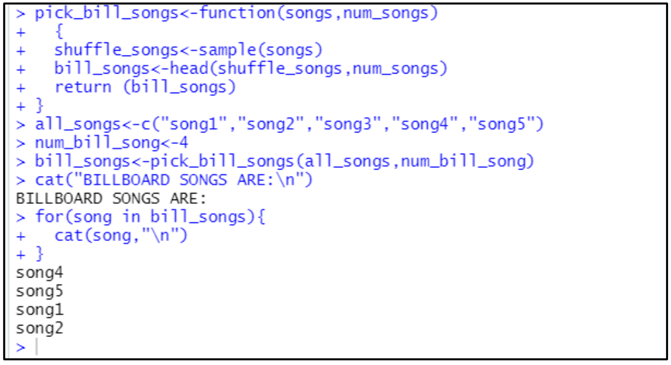
```
C:\Users\DELL\PycharmProjects\SemIII\venv\Scripts\python.exe C:\Users\DELL\PycharmProjects\SemIII\Songs_StaticData.py
Shravya Erabathini,07
Billboard songs are:
song1
song3
song5

Process finished with exit code 0
```

(C) Using R with Static Data

Code:

```
pick_bill_songs<-function(songs,num_songs)
  {
  shuffle_songs<-sample(songs)
  bill_songs<-head(shuffle_songs,num_songs)
  return (bill_songs)
}
all_songs<-c("song1","song2","song3","song4","song5")
num_bill_song<-4
bill_songs<-pick_bill_songs(all_songs,num_bill_song)
cat("BILLBOARD SONGS ARE:\n")
for(song in bill_songs){
  cat(song,"\n")
}
```

Output:



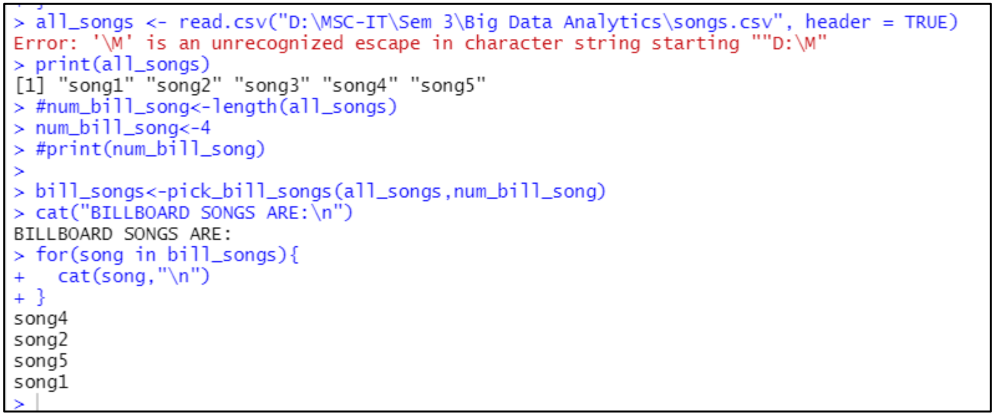
```
> pick_bill_songs<-function(songs,num_songs)
+   {
+   shuffle_songs<-sample(songs)
+   bill_songs<-head(shuffle_songs,num_songs)
+   return (bill_songs)
+ }
> all_songs<-c("song1","song2","song3","song4","song5")
> num_bill_song<-4
> bill_songs<-pick_bill_songs(all_songs,num_bill_song)
> cat("BILLBOARD SONGS ARE:\n")
BILLBOARD SONGS ARE:
> for(song in bill_songs){
+   cat(song,"\n")
+ }
song4
song5
song1
song2
> 
```

(D) Using R with CSV

Code:

```
pick_bill_songs<-function(songs,num_songs)
{
  shuffle_songs<-sample(songs)
  bill_songs<-head(shuffle_songs,num_songs)
  return (bill_songs)
}
all_songs <- read.csv("D:\MSC-IT\Sem 3\Big Data Analytics\songs.csv", header = TRUE)
print(all_songs)
#num_bill_song<-length(all_songs)
num_bill_song<-4
#print(num_bill_song)

bill_songs<-pick_bill_songs(all_songs,num_bill_song)
cat("BILLBOARD SONGS ARE:\n")
for(song in bill_songs){
  cat(song,"\n")
}
```

Output:

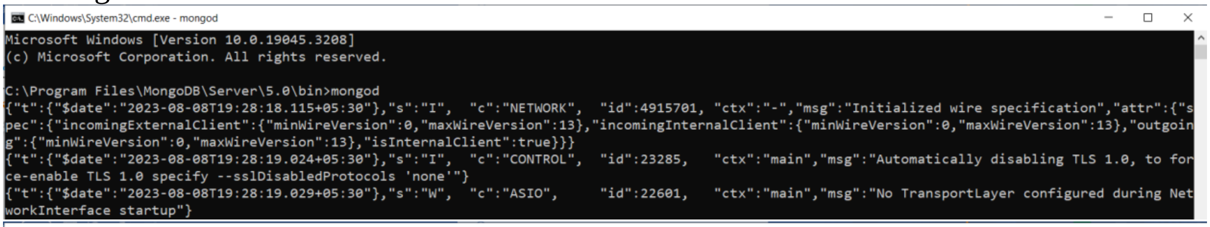


```
> all_songs <- read.csv("D:\MSC-IT\Sem 3\Big Data Analytics\songs.csv", header = TRUE)
Error: '\M' is an unrecognized escape in character string starting ""D:\M"
> print(all_songs)
[1] "song1" "song2" "song3" "song4" "song5"
> #num_bill_song<-length(all_songs)
> num_bill_song<-4
> #print(num_bill_song)
>
> bill_songs<-pick_bill_songs(all_songs,num_bill_song)
> cat("BILLBOARD SONGS ARE:\n")
BILLBOARD SONGS ARE:
> for(song in bill_songs){
+   cat(song,"\n")
+ }
song4
song2
song5
song1
> 
```

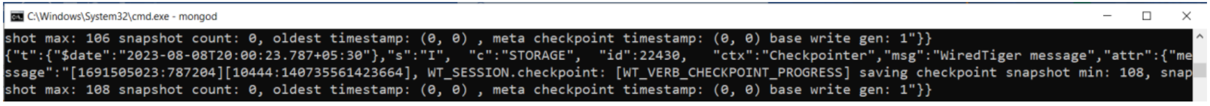
Practical 03

Aim: Implement an application that stores Big Data in Mongo DB and manipulate using R and Python

Step 1: Check whether server is running using mongod command in the cmd at location till bin of mongodb

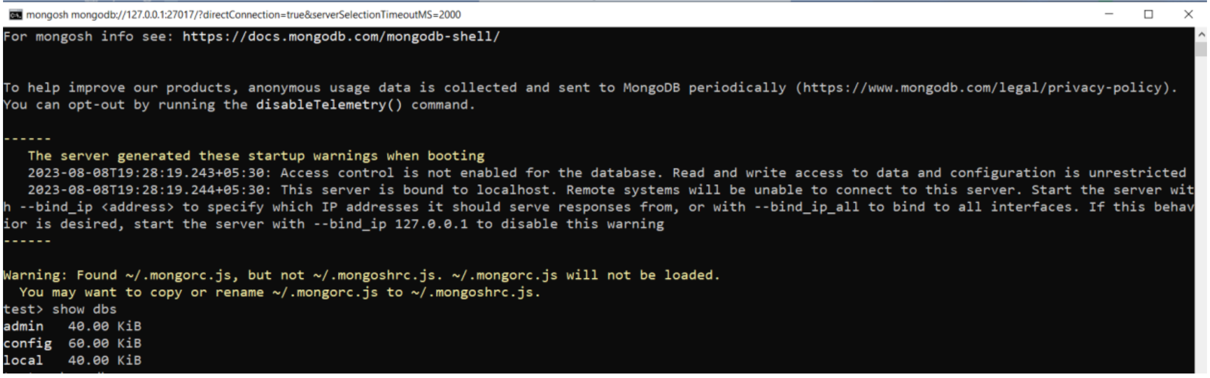


```
Microsoft Windows [Version 10.0.19045.3208]
(c) Microsoft Corporation. All rights reserved.

C:\Program Files\MongoDB\Server\5.0\bin>mongod
{"t":{"$date":"2023-08-08T19:28:18.115+05:30"},"s":"I",  "c":"NETWORK",  "id":4915701, "ctx":"-","msg":"Initialized wire specification","attr":{"s
pec":{"incomingExternalClient":{"minWireVersion":0,"maxWireVersion":13},"incomingInternalClient":{"minWireVersion":0,"maxWireVersion":13},"outgoin
g":{"minWireVersion":0,"maxWireVersion":13},"isInternalClient":true}}}
{"t":{"$date":"2023-08-08T19:28:19.024+05:30"},"s":"I",  "c":"CONTROL",  "id":23285,   "ctx":"main","msg":"Automatically disabling TLS 1.0, to for
ce-enable TLS 1.0 specify --sslDisabledProtocols 'none'"}
{"t":{"$date":"2023-08-08T19:28:19.029+05:30"},"s":"W",  "c":"ASIO",     "id":22601,   "ctx":"main","msg":"No TransportLayer configured during Net
workInterface startup"}
```



```
shot max: 106 snapshot count: 0, oldest timestamp: (0, 0) , meta checkpoint timestamp: (0, 0) base write gen: 1"}}
{"t":{"$date":"2023-08-08T20:00:23.787+05:30"},"s":"I",  "c":"STORAGE",  "id":22430,   "ctx":"Checkpointer","msg":"WiredTiger message","attr":{"me
ssage":"[1691505023:787204][10444:140735561423664], WT_SESSION.checkpoint: [WT_VERB_CHECKPOINT_PROGRESS] saving checkpoint snapshot min: 108, snap
shot max: 108 snapshot count: 0, oldest timestamp: (0, 0) , meta checkpoint timestamp: (0, 0) base write gen: 1"}}
```

Step 2: On another shell enter command **mongosh**



mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000

```
For mongosh info see: https://docs.mongodb.com/mongodb-shell/


To help improve our products, anonymous usage data is collected and sent to MongoDB periodically (https://www.mongodb.com/legal/privacy-policy).
You can opt-out by running the disableTelemetry() command.

------
   The server generated these startup warnings when booting
   2023-08-08T19:28:19.243+05:30: Access control is not enabled for the database. Read and write access to data and configuration is unrestricted
   2023-08-08T19:28:19.244+05:30: This server is bound to localhost. Remote systems will be unable to connect to this server. Start the server wit
h --bind_ip <address> to specify which IP addresses it should serve responses from, or with --bind_ip_all to bind to all interfaces. If this behav
ior is desired, start the server with --bind_ip 127.0.0.1 to disable this warning
------

Warning: Found ~/.mongorc.js, but not ~/.mongoshrc.js. ~/.mongorc.js will not be loaded.
  You may want to copy or rename ~/.mongorc.js to ~/.mongoshrc.js.
test> show dbs
admin   40.00 KiB
config  60.00 KiB
local   40.00 KiB
```

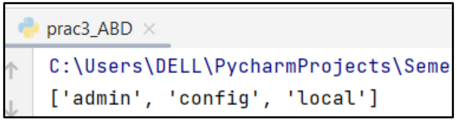
(A) Using Python

Step 1: Establish a connection with Mongodb Create a database “mydatabase”

Code:

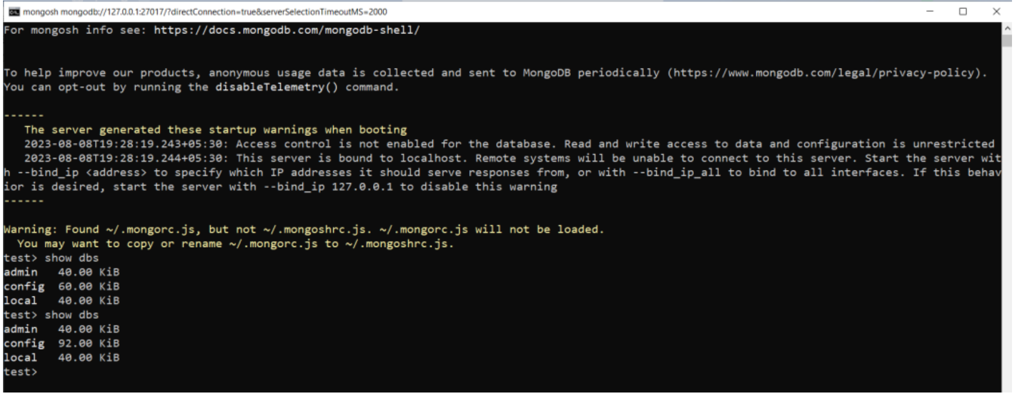
```
import pymongo
myclient=pymongo.MongoClient("mongodb://127.0.0.1:27017/")
mydb=myclient["mydatabase"]
```

Output:



```
C:\Users\DELL\PycharmProjects\Seme
['admin', 'config', 'local']
```

Step 2: Verify in the shell whether database is displayed or not.



mongosh mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000

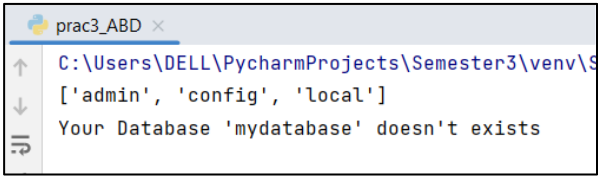
```
For mongosh info see: https://docs.mongodb.com/mongodb-shell/


To help improve our products, anonymous usage data is collected and sent to MongoDB periodically (https://www.mongodb.com/legal/privacy-policy).
You can opt-out by running the disableTelemetry() command.

------
   The server generated these startup warnings when booting
   2023-08-08T19:28:19.243+05:30: Access control is not enabled for the database. Read and write access to data and configuration is unrestricted
   2023-08-08T19:28:19.244+05:30: This server is bound to localhost. Remote systems will be unable to connect to this server. Start the server wit
h --bind_ip <address> to specify which IP addresses it should serve responses from, or with --bind_ip_all to bind to all interfaces. If this behav
ior is desired, start the server with --bind_ip 127.0.0.1 to disable this warning
------

Warning: Found ~/.mongorc.js, but not ~/.mongoshrc.js. ~/.mongorc.js will not be loaded.
  You may want to copy or rename ~/.mongorc.js to ~/.mongoshrc.js.
test> show dbs
admin   40.00 KiB
config  60.00 KiB
local   40.00 KiB
test> show dbs
admin   40.00 KiB
config  92.00 KiB
local   40.00 KiB
test>
```

Step 3: List the database and verify if it exists

Code:

```
import pymongo
myclient=pymongo.MongoClient("mongodb://127.0.0.1:27017/")
mydb=myclient["mydatabase"]
print(myclient.list_database_names())
dblist=myclient.list_database_names()
if "mydatabase" in dblist:
    print("Your Database 'mydatabase' exist")
else:
    print("Your Database 'mydatabase' doesn't exists")
```

Output:



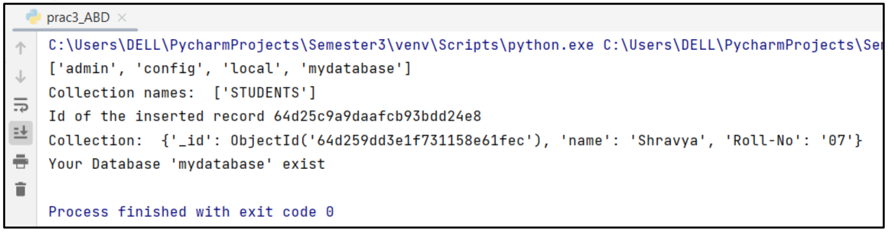
```
C:\Users\DELL\PycharmProjects\Semester3\venv\S
['admin', 'config', 'local']
Your Database 'mydatabase' doesn't exists
```

Step 4: Enter One collection at a time

Code:

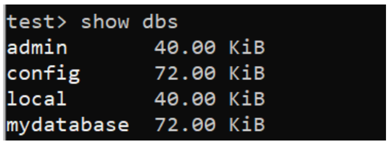
```
import pymongo
myclient=pymongo.MongoClient("mongodb://127.0.0.1:27017/")
mydb=myclient["mydatabase"]
print(myclient.list_database_names())
dblist=myclient.list_database_names()

mycol=mydb["STUDENTS"]  #Collection name
print("Collection names: ",mydb.list_collection_names())

mydict = {"name":"Shravya","Roll-No":"07"}
x=mycol.insert_one(mydict)
print('Id of the inserted record',x.inserted_id)
y=mycol.find_one()
print("Collection: ",y)
if "mydatabase" in dblist:
    print("Your Database 'mydatabase' exist")
else:
    print("Your Database 'mydatabase' doesn't exists")
```

Output:



```
C:\Users\DELL\PycharmProjects\Semester3\venv\Scripts\python.exe C:\Users\DELL\PycharmProjects\Se
['admin', 'config', 'local', 'mydatabase']
Collection names:  ['STUDENTS']
Id of the inserted record 64d25c9a9daafcb93bdd24e8
Collection:  {'_id': ObjectId('64d259dd3e1f731158e61fec'), 'name': 'Shravya', 'Roll-No': '07'}
Your Database 'mydatabase' exist

Process finished with exit code 0
```

Note: Once collections created in the database, It starts reflecting the list of database.



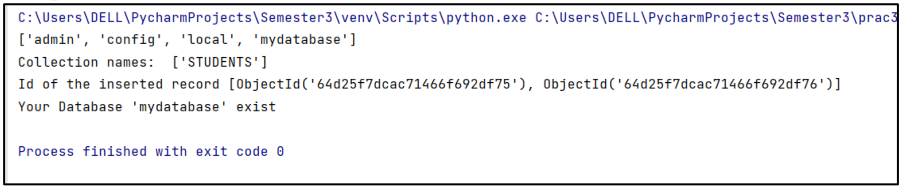
```
test> show dbs
admin       40.00 KiB
config      72.00 KiB
local       40.00 KiB
mydatabase  72.00 KiB
```

Step 5: Enter Multiple collections at a time.

Code:

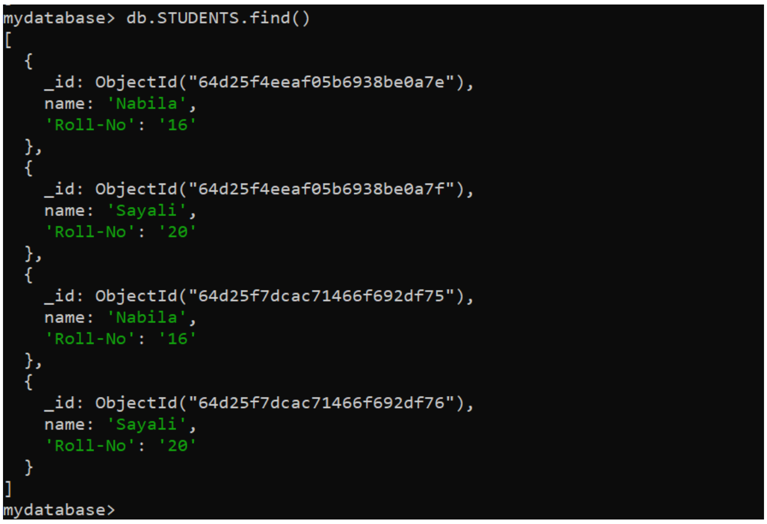
```
import pymongo
myclient=pymongo.MongoClient("mongodb://127.0.0.1:27017/")
mydb=myclient["mydatabase"]
print(myclient.list_database_names())
dblist=myclient.list_database_names()

mycol=mydb['STUDENTS']  #Collection name
print("Collection names: ",mydb.list_collection_names())
mydict = {"name":"Nabila","Roll-No":"16"},{"name":"Sayali","Roll-No":"20"}
x=mycol.insert_many(mydict)
print('Id of the inserted record',x.inserted_ids)
#for i in mycol.find():
   #print("Collections: ",i)
if "mydatabase" in dblist:
   print("Your Database 'mydatabase' exist")
else:
   print("Your Database 'mydatabase' doesn't exists")
```

Output:



```
C:\Users\DELL\PycharmProjects\Semester3\venv\Scripts\python.exe C:\Users\DELL\PycharmProjects\Semester3\prac3
['admin', 'config', 'local', 'mydatabase']
Collection names:  ['STUDENTS']
Id of the inserted record [ObjectId('64d25f7dcac71466f692df75'), ObjectId('64d25f7dcac71466f692df76')]
Your Database 'mydatabase' exist

Process finished with exit code 0
```

1) Displaying the collections of database in mongo shell



```
mydatabase> db.STUDENTS.find()
[
  {
    _id: ObjectId("64d25f4eeaf05b6938be0a7e"),
    name: 'Nabila',
    'Roll-No': '16'
  },
  {
    _id: ObjectId("64d25f4eeaf05b6938be0a7f"),
    name: 'Sayali',
    'Roll-No': '20'
  },
  {
    _id: ObjectId("64d25f7dcac71466f692df75"),
    name: 'Nabila',
    'Roll-No': '16'
  },
  {
    _id: ObjectId("64d25f7dcac71466f692df76"),
    name: 'Sayali',
    'Roll-No': '20'
  }
]
mydatabase>
```

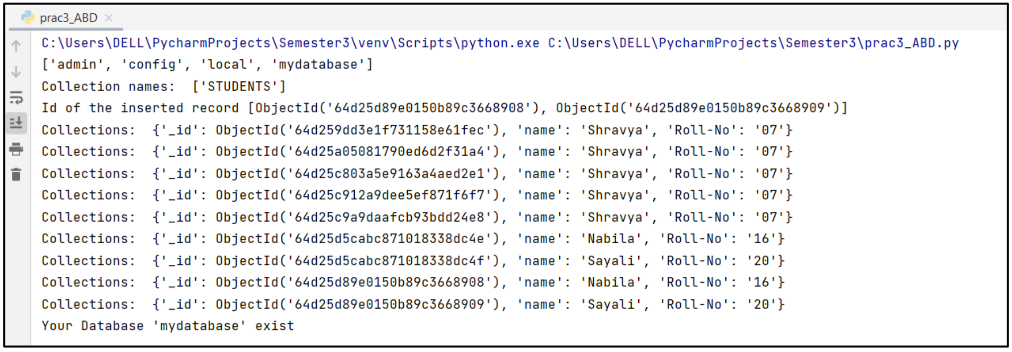
2) Displaying the collections of database in python shell

Code:

```
import pymongo
myclient=pymongo.MongoClient(“mongodb://127.0.0.1:27017/”)
mydb=myclient[“mydatabase”]
print(myclient.list_database_names())
dblist=myclient.list_database_names()

mycol=mydb[“STUDENTS”]  #Collection name
print(“Collection names: “,mydb.list_collection_names())
```

```
mydict = {“name”:”Nabila”,”Roll-No”:”16”},{“name”:”Sayali”,”Roll-No”:”20”}
x=mycol.insert_many(mydict)
print(‘Id of the inserted record’,x.inserted_ids)
for I in mycol.find():
   print(“Collections: “,i)

if “mydatabase” in dblist:
   print(“Your Database ‘mydatabase’ exist”)
else:
   print(“Your Database ‘mydatabase’ doesn’t exists”)
```

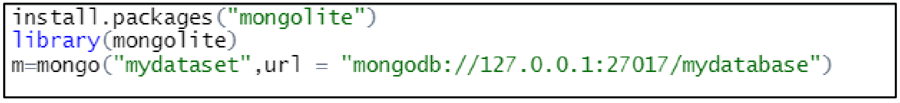
Output:



```
C:\Users\DELL\PycharmProjects\Semester3\venv\Scripts\python.exe C:\Users\DELL\PycharmProjects\Semester3\prac3_ABD.py
['admin', 'config', 'local', 'mydatabase']
Collection names:  ['STUDENTS']
Id of the inserted record [ObjectId('64d25d89e0150b89c3668908'), ObjectId('64d25d89e0150b89c3668909')]
Collections:  {'_id': ObjectId('64d259dd3e1f731158e61fec'), 'name': 'Shravya', 'Roll-No': '07'}
Collections:  {'_id': ObjectId('64d25a05081790ed6d2f31a4'), 'name': 'Shravya', 'Roll-No': '07'}
Collections:  {'_id': ObjectId('64d25c803a5e9163a4aed2e1'), 'name': 'Shravya', 'Roll-No': '07'}
Collections:  {'_id': ObjectId('64d25c912a9dee5ef871f6f7'), 'name': 'Shravya', 'Roll-No': '07'}
Collections:  {'_id': ObjectId('64d25c9a9daafcb93bdd24e8'), 'name': 'Shravya', 'Roll-No': '07'}
Collections:  {'_id': ObjectId('64d25d5cabc871018338dc4e'), 'name': 'Nabila', 'Roll-No': '16'}
Collections:  {'_id': ObjectId('64d25d5cabc871018338dc4f'), 'name': 'Sayali', 'Roll-No': '20'}
Collections:  {'_id': ObjectId('64d25d89e0150b89c3668908'), 'name': 'Nabila', 'Roll-No': '16'}
Collections:  {'_id': ObjectId('64d25d89e0150b89c3668909'), 'name': 'Sayali', 'Roll-No': '20'}
Your Database 'mydatabase' exist
```

(B) Using R

Step 1: Connecting R with Mongodb

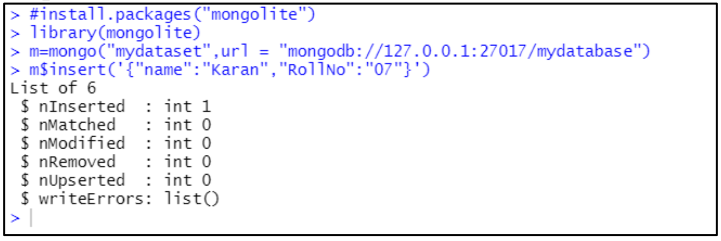


```
install.packages("mongolite")
library(mongolite)
m=mongo("mydataset",url = "mongodb://127.0.0.1:27017/mydatabase")
```

Step 2: Inserting values

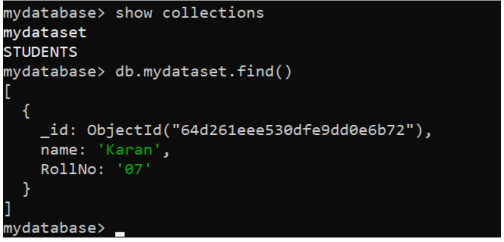
```
#install.packages("mongolite")
library(mongolite)
m=mongo("mydataset",url = "mongodb://127.0.0.1:27017/mydatabase")
m$insert('{"name":"Karan","RollNo":"07"}')
```

Output:



```
> #install.packages("mongolite")
> library(mongolite)
> m=mongo("mydataset",url = "mongodb://127.0.0.1:27017/mydatabase")
> m$insert('{"name":"Karan","RollNo":"07"}')
List of 6
 $ nInserted  : int 1
 $ nMatched   : int 0
 $ nModified  : int 0
 $ nRemoved   : int 0
 $ nUpserted  : int 0
 $ writeErrors: list()
> 
```

Step 3: Verifying in console



```
mydatabase> show collections
mydataset
STUDENTS
mydatabase> db.mydataset.find()
[
  {
    _id: ObjectId("64d261eee530dfe9dd0e6b72"),
    name: 'Karan',
    RollNo: '07'
  }
]
mydatabase> _
```

Signature

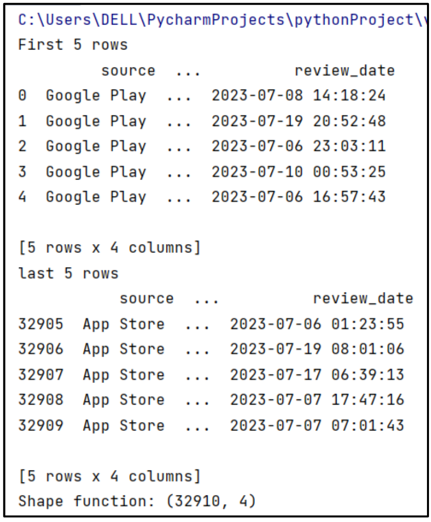
Practical 04

4A) Aim: Analyzing instagram app reviews

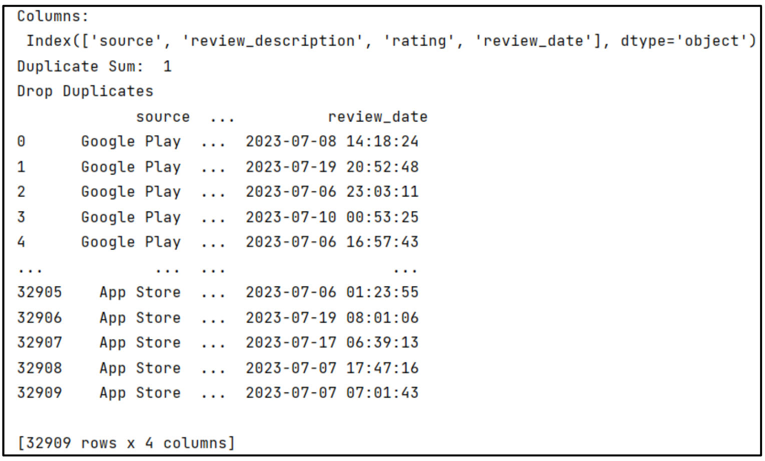
Code:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import warnings
import seaborn as sns
warnings.filterwarnings('ignore')

df=pd.read_csv("threads_reviews.csv")
print("First 5 rows\n",df.head())
print("last 5 rows\n",df.tail())
print("Shape function:",df.shape)
print("Columns:\n",df.columns)
print("Duplicate Sum: ",df.duplicated().sum())
print("Drop Duplicates\n",df.drop_duplicates())
print("IsNull sum\n",df.isnull().sum())
print("Information:\n",df.info)
print("Describe\n",df.describe())
print("Unique values\n",df.nunique())
print("Unique with respect to Source",df['source'].unique())
plt.figure(figsize=(15,7))

sns.countplot(x='source',data=df,palette='hls')
plt.show()

plt.figure(figsize=(12,12))
counts=df['source'].value_counts()
plt.pie(counts,labels=counts.index,autopct="%1.1f%%",colors=sns.color_palette('hls'))
plt.title("Source of reviews")
plt.show()
```

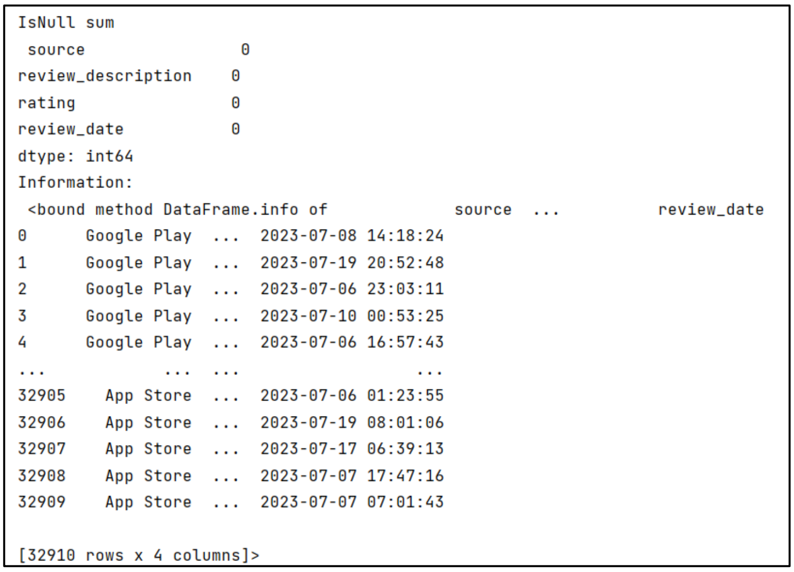
Output:



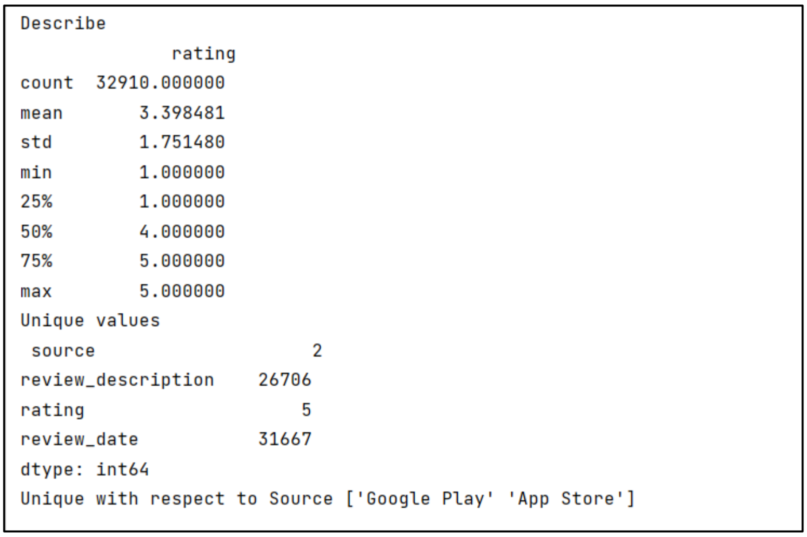
```
C:\Users\DELL\PycharmProjects\pythonProject\
First 5 rows
         source  ...          review_date
0  Google Play  ...  2023-07-08 14:18:24
1  Google Play  ...  2023-07-19 20:52:48
2  Google Play  ...  2023-07-06 23:03:11
3  Google Play  ...  2023-07-10 00:53:25
4  Google Play  ...  2023-07-06 16:57:43

[5 rows x 4 columns]
last 5 rows
             source  ...          review_date
32905  App Store  ...  2023-07-06 01:23:55
32906  App Store  ...  2023-07-19 08:01:06
32907  App Store  ...  2023-07-17 06:39:13
32908  App Store  ...  2023-07-07 17:47:16
32909  App Store  ...  2023-07-07 07:01:43

[5 rows x 4 columns]
Shape function: (32910, 4)
```



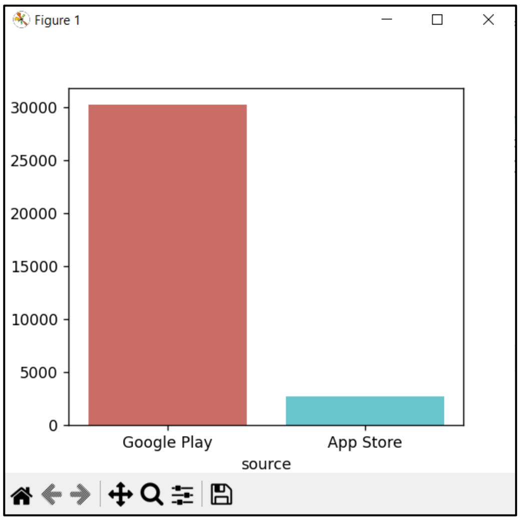
```
Columns:
 Index(['source', 'review_description', 'rating', 'review_date'], dtype='object')
Duplicate Sum:  1
Drop Duplicates
             source  ...          review_date
0      Google Play  ...  2023-07-08 14:18:24
1      Google Play  ...  2023-07-19 20:52:48
2      Google Play  ...  2023-07-06 23:03:11
3      Google Play  ...  2023-07-10 00:53:25
4      Google Play  ...  2023-07-06 16:57:43
...          ...  ...                  ...
32905    App Store  ...  2023-07-06 01:23:55
32906    App Store  ...  2023-07-19 08:01:06
32907    App Store  ...  2023-07-17 06:39:13
32908    App Store  ...  2023-07-07 17:47:16
32909    App Store  ...  2023-07-07 07:01:43

[32909 rows x 4 columns]
```

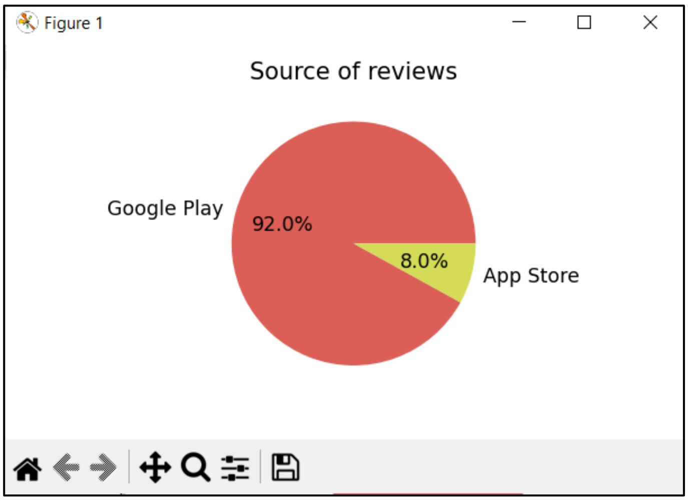


```
IsNull sum
 source                0
review_description    0
rating                0
review_date           0
dtype: int64
Information:
 <bound method DataFrame.info of             source  ...          review_date
0      Google Play  ...  2023-07-08 14:18:24
1      Google Play  ...  2023-07-19 20:52:48
2      Google Play  ...  2023-07-06 23:03:11
3      Google Play  ...  2023-07-10 00:53:25
4      Google Play  ...  2023-07-06 16:57:43
...            ...  ...                  ...
32905    App Store  ...  2023-07-06 01:23:55
32906    App Store  ...  2023-07-19 08:01:06
32907    App Store  ...  2023-07-17 06:39:13
32908    App Store  ...  2023-07-07 17:47:16
32909    App Store  ...  2023-07-07 07:01:43

[32910 rows x 4 columns]>
```



```
Describe
              rating
count  32910.000000
mean       3.398481
std        1.751480
min        1.000000
25%        1.000000
50%        4.000000
75%        5.000000
max        5.000000
Unique values
 source                    2
review_description    26706
rating                    5
review_date           31667
dtype: int64
Unique with respect to Source ['Google Play' 'App Store']
```



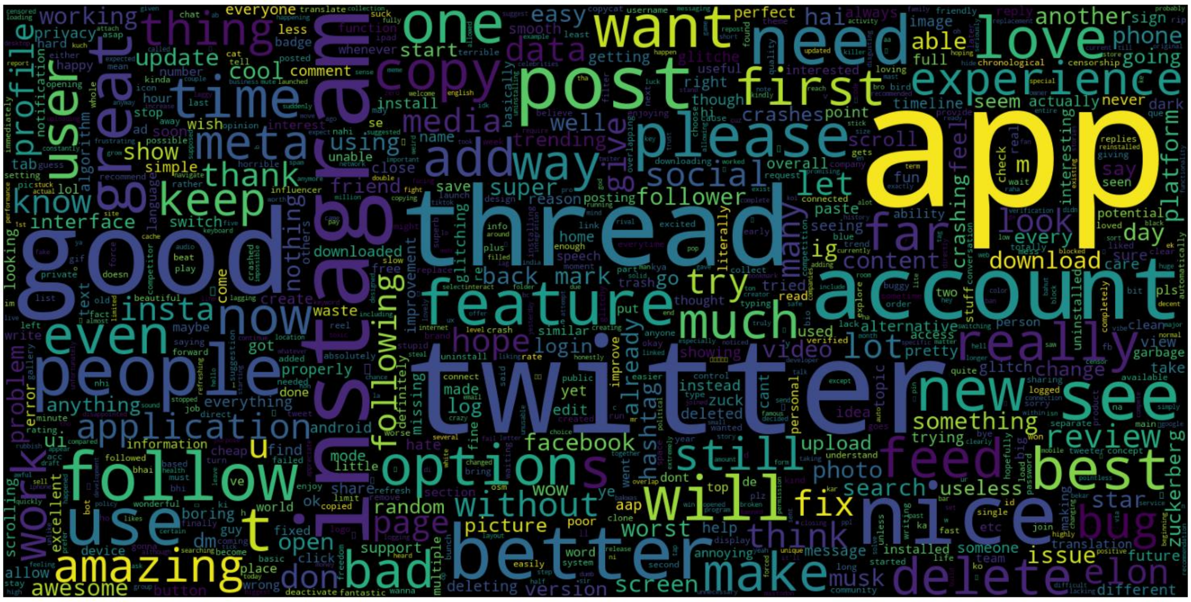


4B) Aim: Data visualization of any social media post with the help of Word Cloud.

Code:

```
import pandas as pd
import wordcloud
from wordcloud import WordCloud
import matplotlib.pyplot as plt

df = pd.read_csv("threads_reviews.csv")
df_new = df[['review_description', 'rating']]
print(df_new)


def clean_text(text):
   text = text.lower()
   return text.strip()


df_new.review_description = df_new.review_description.apply(lambda x: clean_text(x))
print(df_new.review_description)

data=df_new.review_description
plt.figure(figsize=(20,20))
wc=WordCloud(max_words=900,width=1600,height=800,collocations=False).generate(" ".join(data))
plt.imshow(wc)
plt.axis('off')
plt.show()
```

Output:



Signature

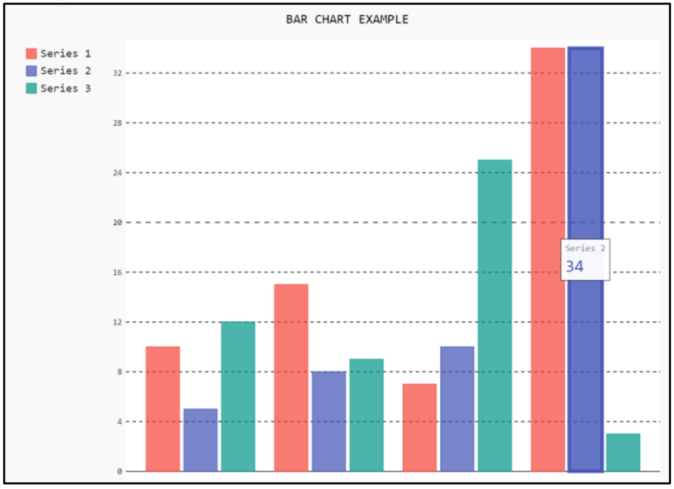
Practical 05

Aim: Data visualization using pygal

Code: BAR CHART

```
import pygal
bar_chart=pygal.Bar()
bar_chart.add("Series 1",[10,15,7,35])
bar_chart.add("Series 2",[5,8,10,34])
bar_chart.add("Series 3",[12,9,25,3])
bar_chart.title="BAR CHART EXAMPLE"
bar_chart.render_to_file("BAR.svg")
```

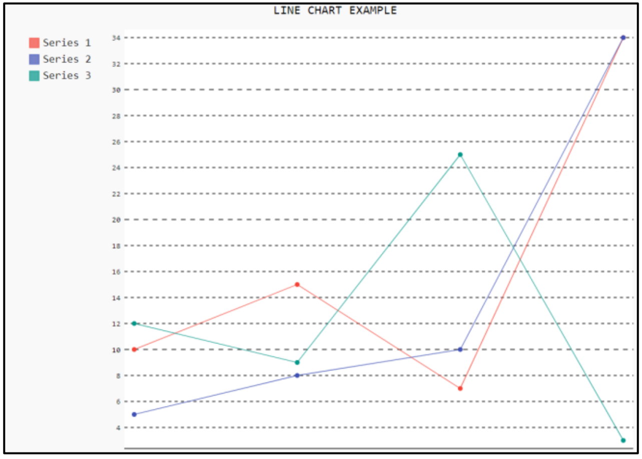
Output:



Code: LINE CHART

```
import pygal
line_chart=pygal.Line()
line_chart.add("Series 1",[10,15,7,35])
line_chart.add("Series 2",[5,8,10,34])
line_chart.add("Series 3",[12,9,25,3])
line_chart.title="LINE CHART EXAMPLE"
line_chart.render_to_file("LINE.svg")
```

Output:

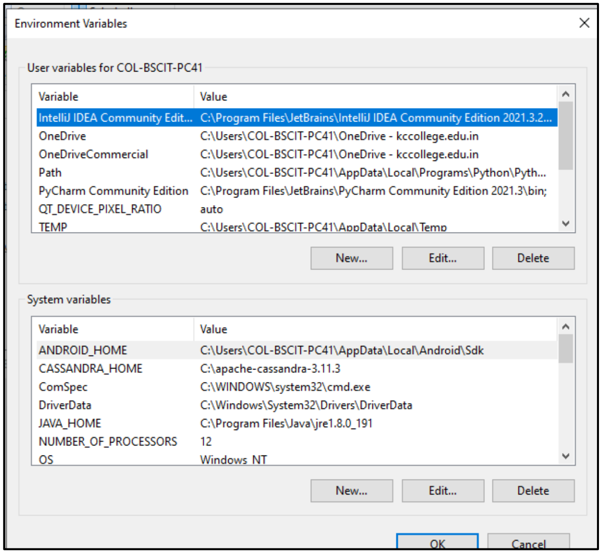


Practical 06

Aim: Installation of Apache Cassandra

1) Install the following:
Python 2.7 , Java 1.8, Cassandra 3.11

2) In the environment variables go to System variables set the path for java as JAVA_HOME (jre path) and cassandra as CASSANDRA_HOME as shown below



Environment Variables

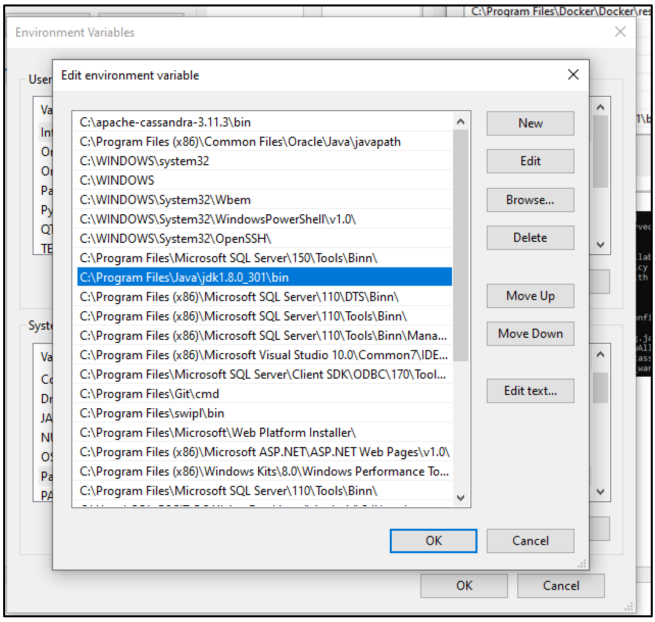
User variables for COL-BSCIT-PC41

Variable	Value
IntelliJ IDEA Community Edit...	C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2021.3.2...
OneDrive	C:\Users\COL-BSCIT-PC41\OneDrive - kccollege.edu.in
OneDriveCommercial	C:\Users\COL-BSCIT-PC41\OneDrive - kccollege.edu.in
Path	C:\Users\COL-BSCIT-PC41\AppData\Local\Programs\Python\Pyth...
PyCharm Community Edition	C:\Program Files\JetBrains\PyCharm Community Edition 2021.3\bin;
QT_DEVICE_PIXEL_RATIO	auto
TEMP	C:\Users\COL-BSCIT-PC41\AppData\Local\Temp

System variables

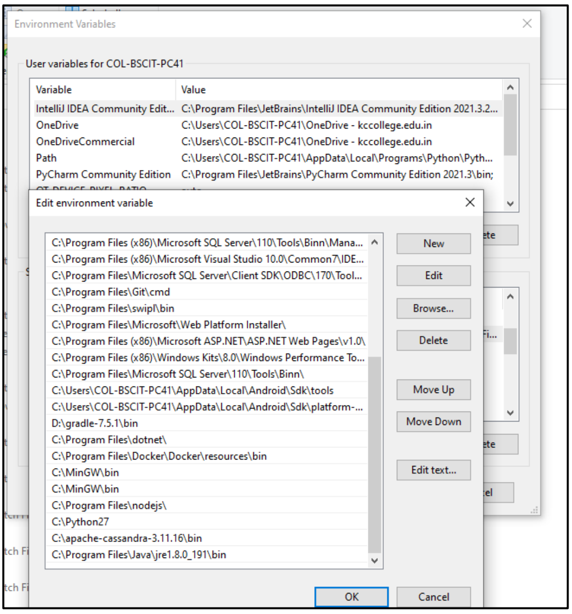
Variable	Value
ANDROID_HOME	C:\Users\COL-BSCIT-PC41\AppData\Local\Android\Sdk
CASSANDRA_HOME	C:\apache-cassandra-3.11.3
ComSpec	C:\WINDOWS\system32\cmd.exe
DriverData	C:\Windows\System32\Drivers\DriverData
JAVA_HOME	C:\Program Files\Java\jre1.8.0_191
NUMBER_OF_PROCESSORS	12
OS	Windows_NT

3) In System Variables go to path -> set the path for java (both jdk and jre path), python and cassandra as shown below



Edit environment variable

```
C:\apache-cassandra-3.11.3\bin
C:\Program Files (x86)\Common Files\Oracle\Java\javapath
C:\WINDOWS\system32
C:\WINDOWS
C:\WINDOWS\System32\Wbem
C:\WINDOWS\System32\WindowsPowerShell\v1.0\
C:\WINDOWS\System32\OpenSSH\
C:\Program Files\Microsoft SQL Server\150\Tools\Binn\
C:\Program Files\Java\jdk1.8.0_301\bin
C:\Program Files (x86)\Microsoft SQL Server\110\DTS\Binn\
C:\Program Files (x86)\Microsoft SQL Server\110\Tools\Binn\
C:\Program Files (x86)\Microsoft SQL Server\110\Tools\Binn\Mana...
C:\Program Files (x86)\Microsoft Visual Studio 10.0\Common7\IDE...
C:\Program Files\Microsoft SQL Server\Client SDK\ODBC\170\Tool...
C:\Program Files\Git\cmd
C:\Program Files\swipl\bin
C:\Program Files\Microsoft\Web Platform Installer\
C:\Program Files (x86)\Microsoft ASP.NET\ASP.NET Web Pages\v1.0\
C:\Program Files (x86)\Windows Kits\8.0\Windows Performance To...
C:\Program Files\Microsoft SQL Server\110\Tools\Binn\
```



Environment Variables

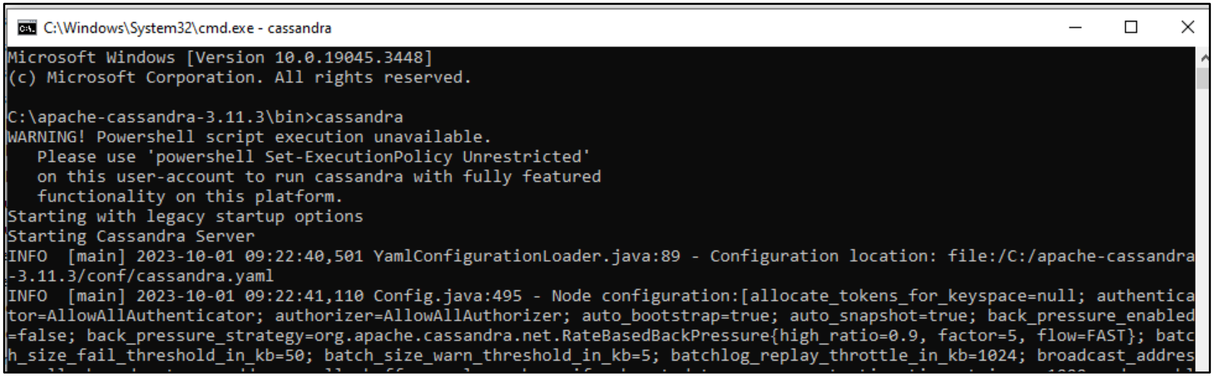
User variables for COL-BSCIT-PC41

Variable	Value
IntelliJ IDEA Community Edit...	C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2021.3.2...
OneDrive	C:\Users\COL-BSCIT-PC41\OneDrive - kccollege.edu.in
OneDriveCommercial	C:\Users\COL-BSCIT-PC41\OneDrive - kccollege.edu.in
Path	C:\Users\COL-BSCIT-PC41\AppData\Local\Programs\Python\Pyth...
PyCharm Community Edition	C:\Program Files\JetBrains\PyCharm Community Edition 2021.3\bin;

Edit environment variable

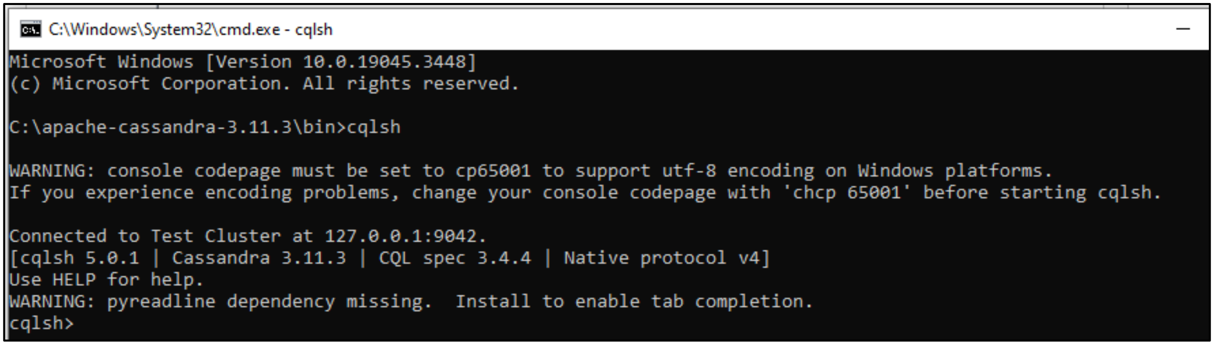
```
C:\Program Files (x86)\Microsoft SQL Server\110\Tools\Binn\Mana...
C:\Program Files (x86)\Microsoft Visual Studio 10.0\Common7\IDE...
C:\Program Files\Microsoft SQL Server\Client SDK\ODBC\170\Tool...
C:\Program Files\Git\cmd
C:\Program Files\swipl\bin
C:\Program Files\Microsoft\Web Platform Installer\
C:\Program Files (x86)\Microsoft ASP.NET\ASP.NET Web Pages\v1.0\
C:\Program Files (x86)\Windows Kits\8.0\Windows Performance To...
C:\Program Files\Microsoft SQL Server\110\Tools\Binn\
C:\Users\COL-BSCIT-PC41\AppData\Local\Android\Sdk\tools
C:\Users\COL-BSCIT-PC41\AppData\Local\Android\Sdk\platform-...
D:\gradle-7.5.1\bin
C:\Program Files\dotnet\
C:\Program Files\Docker\Docker\resources\bin
C:\MinGW\bin
C:\MinGW\bin
C:\Program Files\nodejs\
C:\Python27
C:\apache-cassandra-3.11.16\bin
C:\Program Files\Java\jre1.8.0_191\bin
```

3) Open two Command Prompt -> In fisrt CMD enter the following command

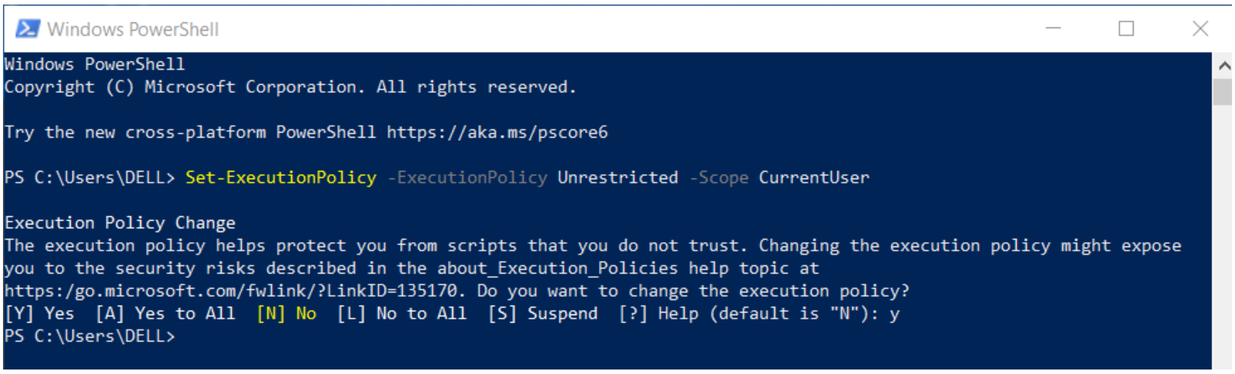


```
Microsoft Windows [Version 10.0.19045.3448]
(c) Microsoft Corporation. All rights reserved.

C:\apache-cassandra-3.11.3\bin>cassandra
WARNING! Powershell script execution unavailable.
   Please use 'powershell Set-ExecutionPolicy Unrestricted'
   on this user-account to run cassandra with fully featured
   functionality on this platform.
Starting with legacy startup options
Starting Cassandra Server
INFO  [main] 2023-10-01 09:22:40,501 YamlConfigurationLoader.java:89 - Configuration location: file:/C:/apache-cassandra
-3.11.3/conf/cassandra.yaml
INFO  [main] 2023-10-01 09:22:41,110 Config.java:495 - Node configuration:[allocate_tokens_for_keyspace=null; authentica
tor=AllowAllAuthenticator; authorizer=AllowAllAuthorizer; auto_bootstrap=true; auto_snapshot=true; back_pressure_enabled
=false; back_pressure_strategy=org.apache.cassandra.net.RateBasedBackPressure{high_ratio=0.9, factor=5, flow=FAST}; batc
h_size_fail_threshold_in_kb=50; batch_size_warn_threshold_in_kb=5; batchlog_replay_throttle_in_kb=1024; broadcast_addres
```

4) In Second CMD enter the command **cqlsh** as shown below



```
Microsoft Windows [Version 10.0.19045.3448]
(c) Microsoft Corporation. All rights reserved.

C:\apache-cassandra-3.11.3\bin>cqlsh

WARNING: console codepage must be set to cp65001 to support utf-8 encoding on Windows platforms.
If you experience encoding problems, change your console codepage with 'chcp 65001' before starting cqlsh.

Connected to Test Cluster at 127.0.0.1:9042.
[cqlsh 5.0.1 | Cassandra 3.11.3 | CQL spec 3.4.4 | Native protocol v4]
Use HELP for help.
WARNING: pyreadline dependency missing.  Install to enable tab completion.
cqlsh>
```



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\DELL> Set-ExecutionPolicy -ExecutionPolicy Unrestricted -Scope CurrentUser

Execution Policy Change
The execution policy helps protect you from scripts that you do not trust. Changing the execution policy might expose
you to the security risks described in the about_Execution_Policies help topic at
https:/go.microsoft.com/fwlink/?LinkID=135170. Do you want to change the execution policy?
[Y] Yes  [A] Yes to All  [N] No  [L] No to All  [S] Suspend  [?] Help (default is "N"): y
PS C:\Users\DELL>
```

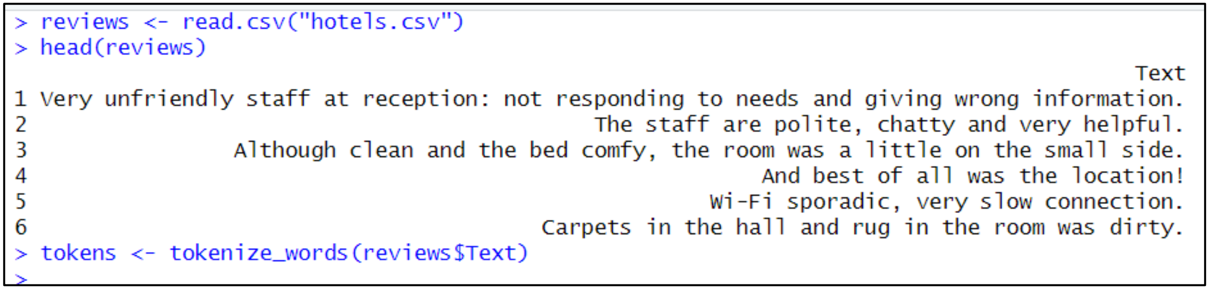
Practical 07

7A) Aim: Sentimental Analysis in R

Code:

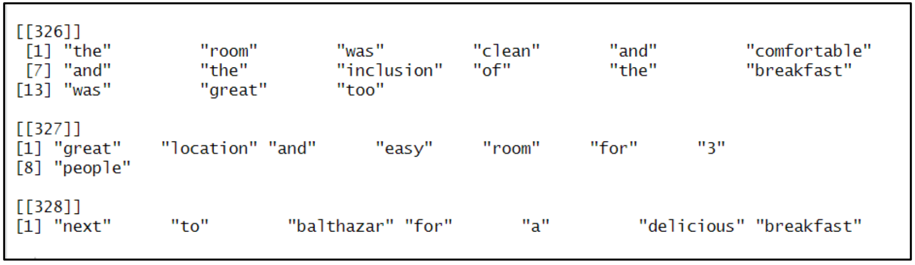
```
install.packages('sentimentr')
install.packages('tidyverse')
install.packages('wordcloud')
install.packages('tokenizers')
install.packages('tm')
install.packages('SentimentalAnalysis')
library(tidyverse)
library(sentimentr)
library(wordcloud)
library(tokenizers)
library(tm)

reviews <- read.csv("hotels.csv")
head(reviews)
tokens <- tokenize_words(reviews$Text)
print(tokens)
wordcloud(words=tokens,min.freq=10,max.words = 50,colors="black")
summary(tokens)
Corpus <-Corpus(VectorSource(reviews))
Corpus <- tm_map(Corpus,removePunctuation)
Corpus <- tm_map(Corpus,removeNumbers)
Corpus <-tm_map(Corpus,content_transformer(tolower))
summary(Corpus)
head(Corpus)
dtm <- DocumentTermMatrix(Corpus)

sentiments <- sentimentr_data(dtm)
for (i in 1:length(sentiments))
{cat (paste("Reviews:" ,reviews[i], "\n"))
  cat (paste ("Sentiment:",sentiments[i], "\n \n"))}
```

Output:



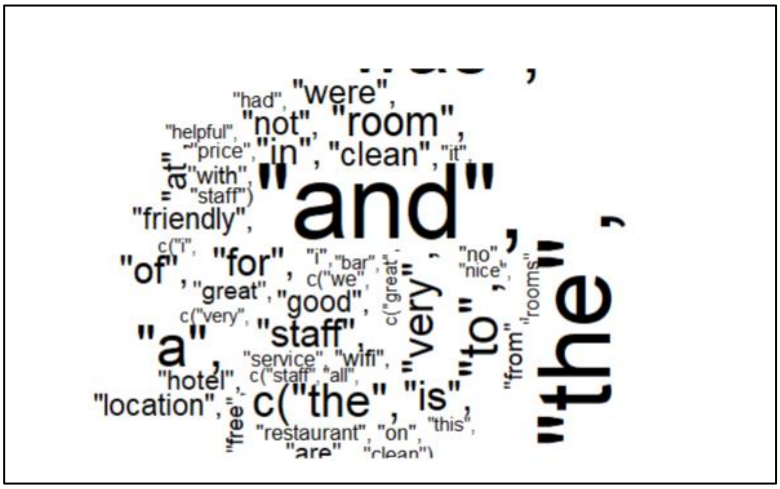
```
> reviews <- read.csv("hotels.csv")
> head(reviews)
                                                                                       Text
1 Very unfriendly staff at reception: not responding to needs and giving wrong information.
2                                            The staff are polite, chatty and very helpful.
3                Although clean and the bed comfy, the room was a little on the small side.
4                                                          And best of all was the location!
5                                                      Wi-Fi sporadic, very slow connection.
6                                         Carpets in the hall and rug in the room was dirty.
> tokens <- tokenize_words(reviews$Text)
>
```

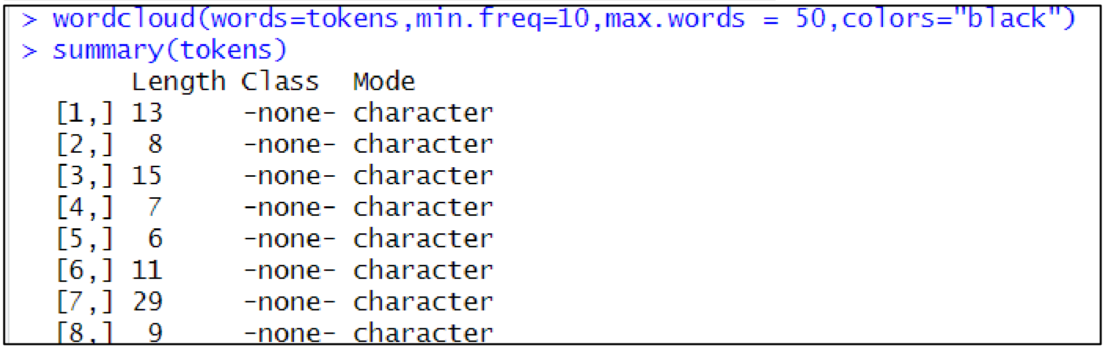
Print(tokens)



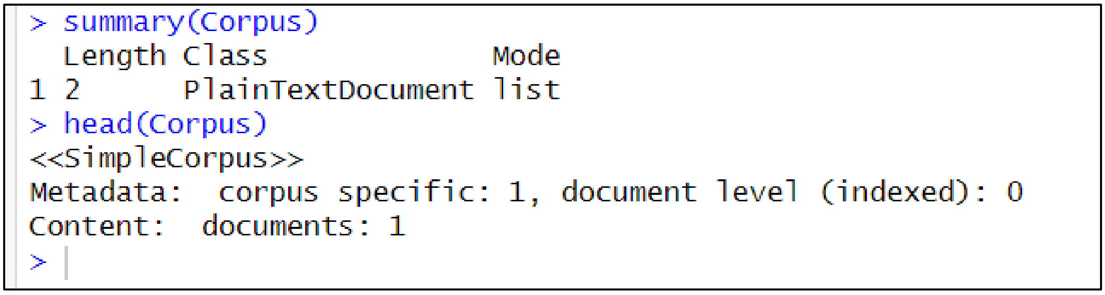
```
[[326]]
 [1] "the"         "room"        "was"         "clean"       "and"         "comfortable"
 [7] "and"         "the"         "inclusion"   "of"          "the"         "breakfast"
[13] "was"         "great"       "too"

[[327]]
[1] "great"    "location" "and"      "easy"     "room"     "for"      "3"
[8] "people"

[[328]]
[1] "next"      "to"        "balthazar" "for"       "a"         "delicious" "breakfast"
```

Signature

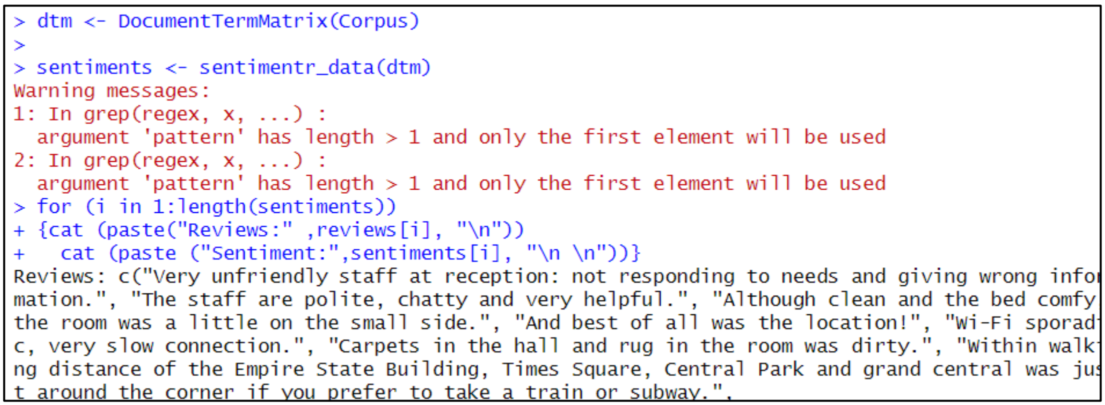




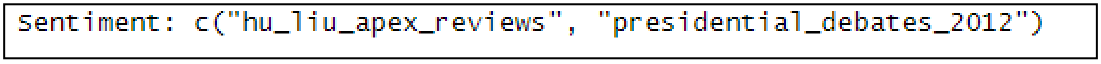
```
> wordcloud(words=tokens,min.freq=10,max.words = 50,colors="black")
> summary(tokens)
      Length Class  Mode
  [1,] 13     -none- character
  [2,]  8     -none- character
  [3,] 15     -none- character
  [4,]  7     -none- character
  [5,]  6     -none- character
  [6,] 11     -none- character
  [7,] 29     -none- character
  [8,]  9     -none- character
```



```
> summary(Corpus)
  Length Class            Mode
1 2      PlainTextDocument list
> head(Corpus)
<<SimpleCorpus>>
Metadata:  corpus specific: 1, document level (indexed): 0
Content:  documents: 1
> 
```



```
> dtm <- DocumentTermMatrix(Corpus)
>
> sentiments <- sentimentr_data(dtm)
Warning messages:
1: In grep(regex, x, ...) :
  argument 'pattern' has length > 1 and only the first element will be used
2: In grep(regex, x, ...) :
  argument 'pattern' has length > 1 and only the first element will be used
> for (i in 1:length(sentiments))
+ {cat (paste("Reviews:" ,reviews[i], "\n"))
+   cat (paste ("Sentiment:",sentiments[i], "\n \n"))}
Reviews: c("Very unfriendly staff at reception: not responding to needs and giving wrong infor
mation.", "The staff are polite, chatty and very helpful.", "Although clean and the bed comfy
the room was a little on the small side.", "And best of all was the location!", "Wi-Fi sporadi
c, very slow connection.", "Carpets in the hall and rug in the room was dirty.", "Within walki
ng distance of the Empire State Building, Times Square, Central Park and grand central was jus
t around the corner if you prefer to take a train or subway.",
```



```
Sentiment: c("hu_liu_apex_reviews", "presidential_debates_2012")
```

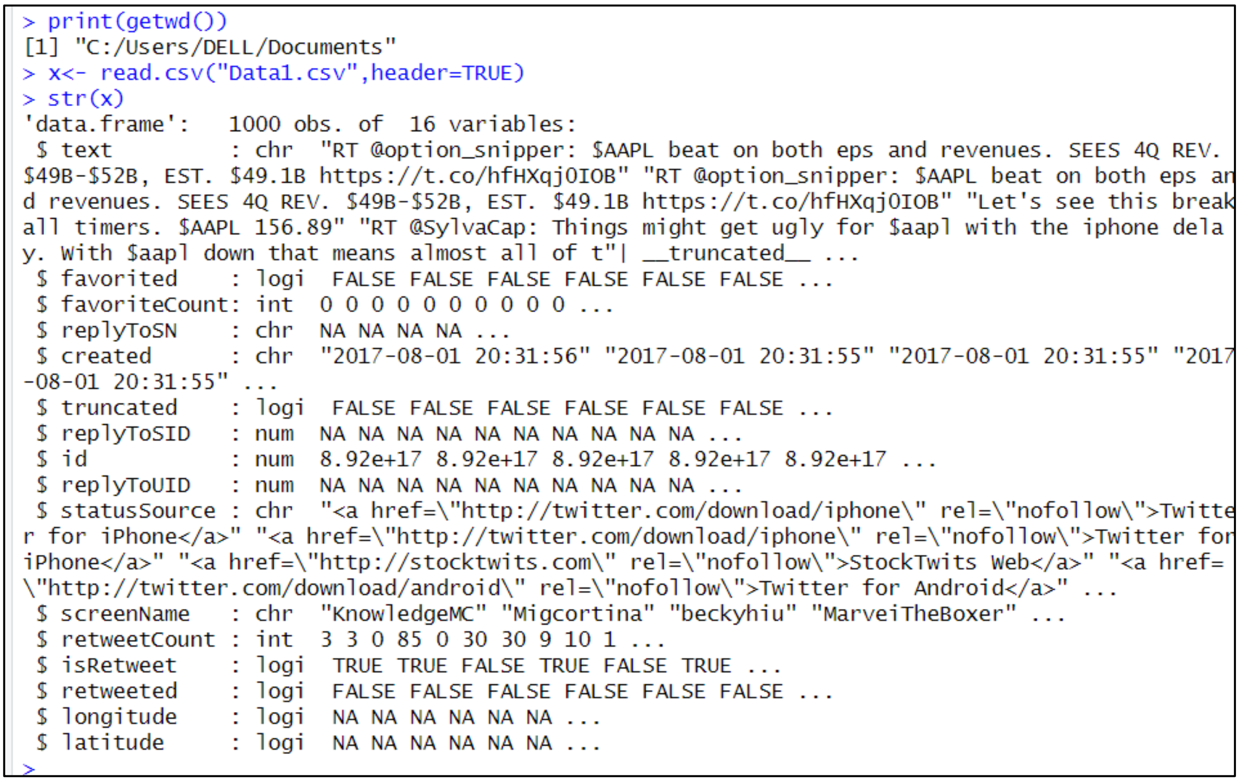
7B) Aim: Sentimental Analysis in R to analyze the data and term document matrix.

Code:

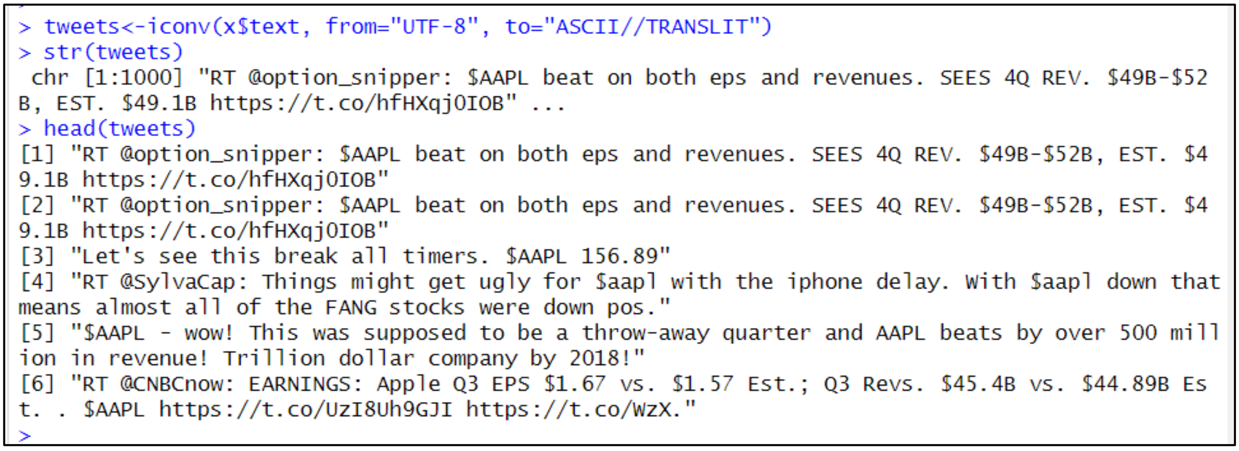
```
install.packages("rlang")
install.packages("syuzhet")
install.packages("lubridate")
install.packages("ggplot2")
install.packages("scales")
install.packages("reshape2")
install.packages("dplyr")

install.packages ("slam")
install.packages("corpus")
library(rlang)
library(syuzhet) #use for sentiment analysis
library(lubridate)
library(ggplot2)
library(scales)
library(reshape2)
library(dplyr)

print(getwd())
x<- read.csv("Data1.csv",header=TRUE)
str(x)

tweets<-iconv(x$text, from="UTF-8", to="ASCII//TRANSLIT")
str(tweets)
head(tweets)

corpus<-iconv(x$text)
corpus<-Corpus(VectorSource(corpus))
inspect(corpus[1:5])
corpus<-tm_map(corpus,tolower)
inspect(corpus[1:5])

corpus<-tm_map(corpus,removeNumbers)
inspect(corpus[1:5])

corpus<-tm_map(corpus,removeWords,stopwords("english"))
inspect(corpus[1:5])

corpus<-tm_map(corpus,removePunctuation)
inspect(corpus[1:5])

tdm<- TermDocumentMatrix(corpus)
tdm<- as.matrix(tdm)
tdm[1:10,1:20]
print(tdm)

a<-rowSums(tdm)
a<-subset(a,a>=45)
barplot(a,col=rainbow(50))
```

Output:

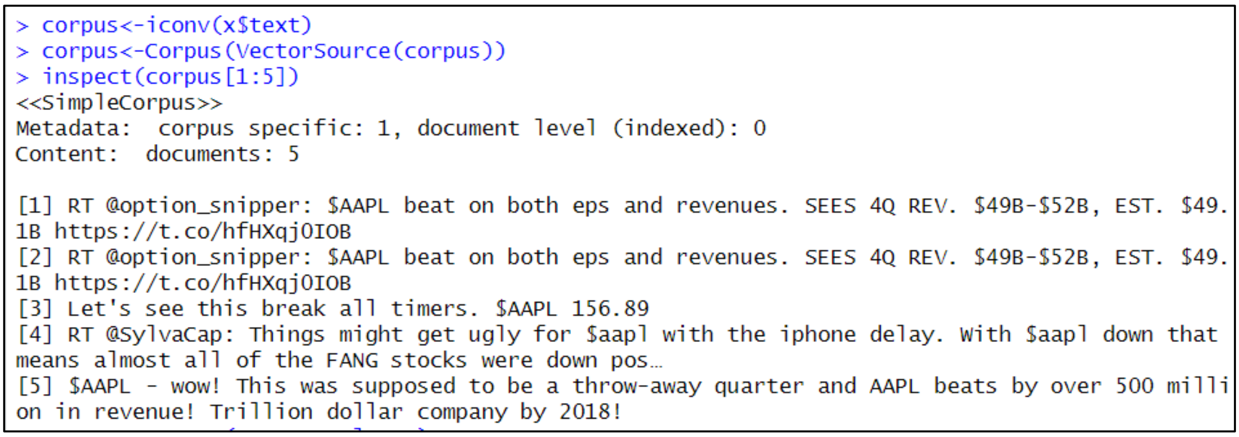
Signature



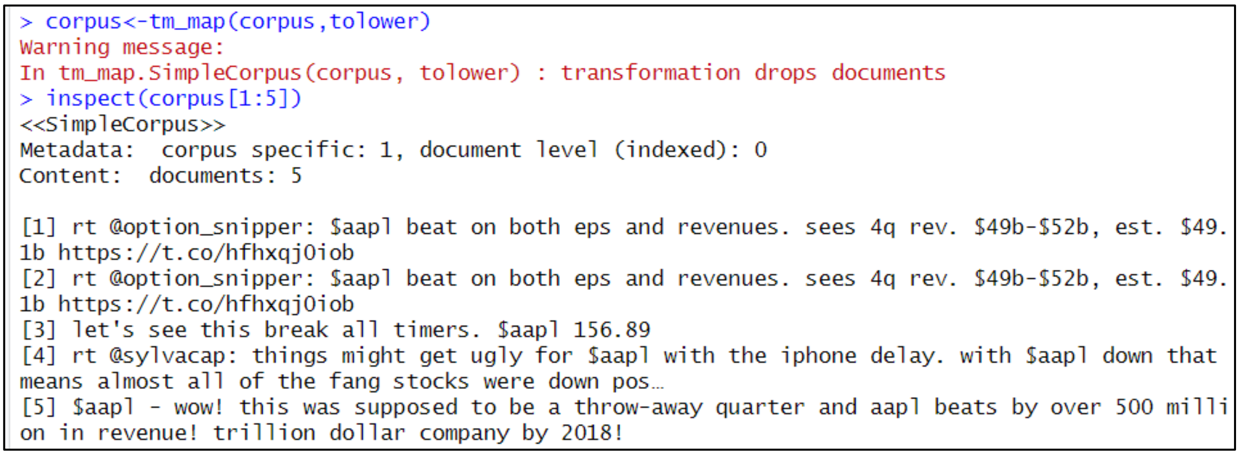
```
> print(getwd())
[1] "C:/Users/DELL/Documents"
> x<- read.csv("Data1.csv",header=TRUE)
> str(x)
'data.frame':	1000 obs. of  16 variables:
 $ text         : chr  "RT @option_snipper: $AAPL beat on both eps and revenues. SEES 4Q REV.
$49B-$52B, EST. $49.1B https://t.co/hfHXqj0IOB" "RT @option_snipper: $AAPL beat on both eps an
d revenues. SEES 4Q REV. $49B-$52B, EST. $49.1B https://t.co/hfHXqj0IOB" "Let's see this break
all timers. $AAPL 156.89" "RT @SylvaCap: Things might get ugly for $aapl with the iphone dela
y. With $aapl down that means almost all of t"| __truncated__ ...
 $ favorited    : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
 $ favoriteCount: int  0 0 0 0 0 0 0 0 0 0 ...
 $ replyToSN    : chr  NA NA NA NA ...
 $ created      : chr  "2017-08-01 20:31:56" "2017-08-01 20:31:55" "2017-08-01 20:31:55" "2017
-08-01 20:31:55" ...
 $ truncated    : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
 $ replyToSID   : num  NA NA NA NA NA NA NA NA NA NA ...
 $ id           : num  8.92e+17 8.92e+17 8.92e+17 8.92e+17 8.92e+17 ...
 $ replyToUID   : num  NA NA NA NA NA NA NA NA NA NA ...
 $ statusSource : chr  "<a href=\"http://twitter.com/download/iphone\" rel=\"nofollow\">Twitte
r for iPhone</a>" "<a href=\"http://twitter.com/download/iphone\" rel=\"nofollow\">Twitter for
iPhone</a>" "<a href=\"http://stocktwits.com\" rel=\"nofollow\">StockTwits Web</a>" "<a href=
\"http://twitter.com/download/android\" rel=\"nofollow\">Twitter for Android</a>" ...
 $ screenName   : chr  "KnowledgeMC" "Migcortina" "beckyhiu" "MarveiTheBoxer" ...
 $ retweetCount : int  3 3 0 85 0 30 30 9 10 1 ...
 $ isRetweet    : logi  TRUE TRUE FALSE TRUE FALSE TRUE ...
 $ retweeted    : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
 $ longitude    : logi  NA NA NA NA NA NA ...
 $ latitude     : logi  NA NA NA NA NA NA ...
>
```



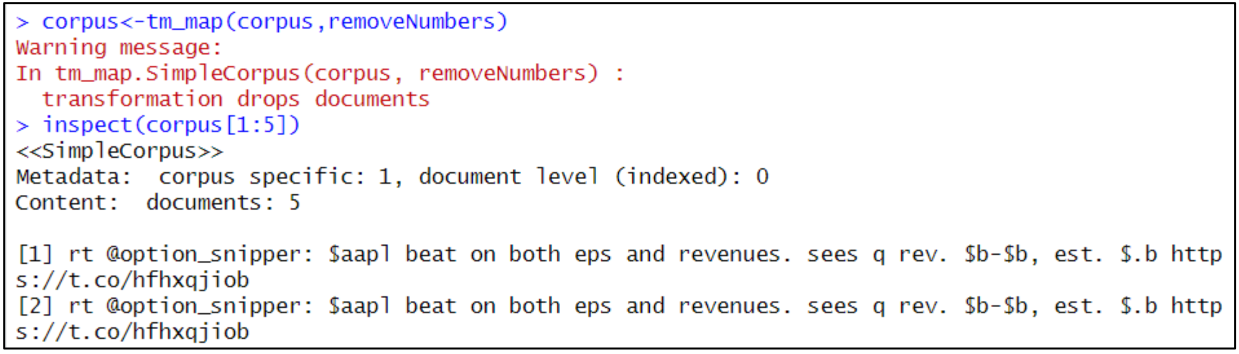
```
> tweets<-iconv(x$text, from="UTF-8", to="ASCII//TRANSLIT")
> str(tweets)
 chr [1:1000] "RT @option_snipper: $AAPL beat on both eps and revenues. SEES 4Q REV. $49B-$52
B, EST. $49.1B https://t.co/hfHXqj0IOB" ...
> head(tweets)
[1] "RT @option_snipper: $AAPL beat on both eps and revenues. SEES 4Q REV. $49B-$52B, EST. $4
9.1B https://t.co/hfHXqj0IOB"
[2] "RT @option_snipper: $AAPL beat on both eps and revenues. SEES 4Q REV. $49B-$52B, EST. $4
9.1B https://t.co/hfHXqj0IOB"
[3] "Let's see this break all timers. $AAPL 156.89"
[4] "RT @SylvaCap: Things might get ugly for $aapl with the iphone delay. With $aapl down that
means almost all of the FANG stocks were down pos."
[5] "$AAPL - wow! This was supposed to be a throw-away quarter and AAPL beats by over 500 mill
ion in revenue! Trillion dollar company by 2018!"
[6] "RT @CNBCnow: EARNINGS: Apple Q3 EPS $1.67 vs. $1.57 Est.; Q3 Revs. $45.4B vs. $44.89B Es
t. . $AAPL https://t.co/UzI8Uh9GJI https://t.co/WzX."
>
```



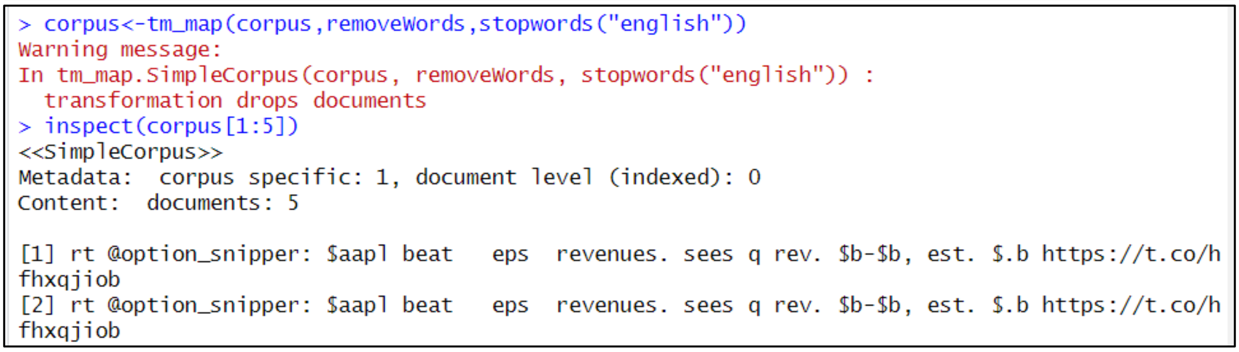
```
> corpus<-iconv(x$text)
> corpus<-Corpus(VectorSource(corpus))
> inspect(corpus[1:5])
<<SimpleCorpus>>
Metadata:  corpus specific: 1, document level (indexed): 0
Content:  documents: 5

[1] RT @option_snipper: $AAPL beat on both eps and revenues. SEES 4Q REV. $49B-$52B, EST. $49.
1B https://t.co/hfHXqj0IOB
[2] RT @option_snipper: $AAPL beat on both eps and revenues. SEES 4Q REV. $49B-$52B, EST. $49.
1B https://t.co/hfHXqj0IOB
[3] Let's see this break all timers. $AAPL 156.89
[4] RT @SylvaCap: Things might get ugly for $aapl with the iphone delay. With $aapl down that
means almost all of the FANG stocks were down pos…
[5] $AAPL - wow! This was supposed to be a throw-away quarter and AAPL beats by over 500 milli
on in revenue! Trillion dollar company by 2018!
```



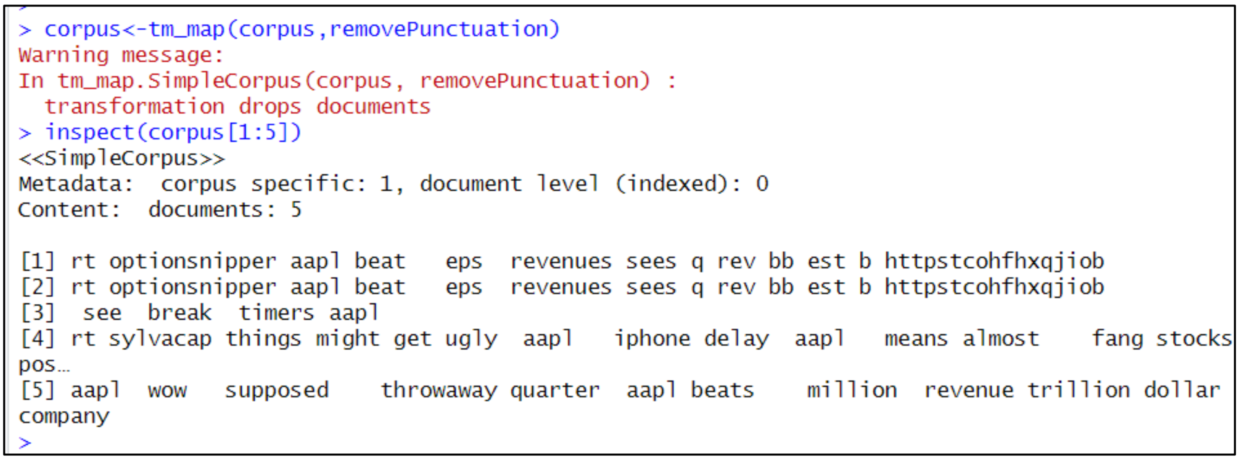
```
> corpus<-tm_map(corpus,tolower)
Warning message:
In tm_map.SimpleCorpus(corpus, tolower) : transformation drops documents
> inspect(corpus[1:5])
<<SimpleCorpus>>
Metadata:  corpus specific: 1, document level (indexed): 0
Content:  documents: 5

[1] rt @option_snipper: $aapl beat on both eps and revenues. sees 4q rev. $49b-$52b, est. $49.
1b https://t.co/hfhxqj0iob
[2] rt @option_snipper: $aapl beat on both eps and revenues. sees 4q rev. $49b-$52b, est. $49.
1b https://t.co/hfhxqj0iob
[3] let's see this break all timers. $aapl 156.89
[4] rt @sylvacap: things might get ugly for $aapl with the iphone delay. with $aapl down that
means almost all of the fang stocks were down pos…
[5] $aapl - wow! this was supposed to be a throw-away quarter and aapl beats by over 500 milli
on in revenue! trillion dollar company by 2018!
```



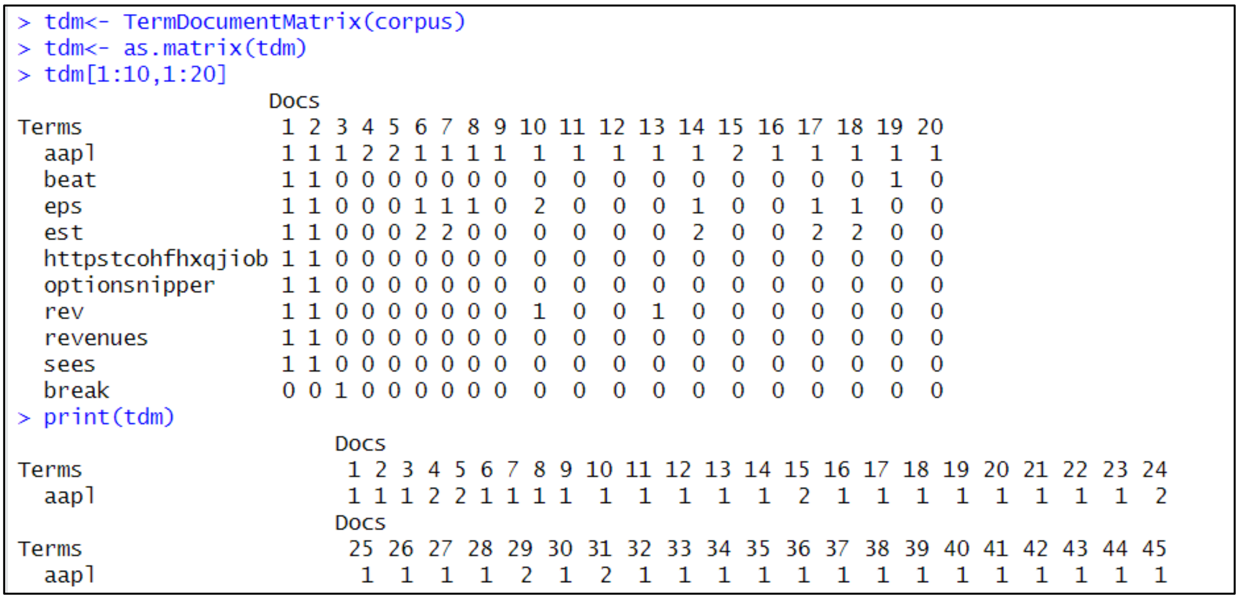
```
> corpus<-tm_map(corpus,removeNumbers)
Warning message:
In tm_map.SimpleCorpus(corpus, removeNumbers) :
  transformation drops documents
> inspect(corpus[1:5])
<<SimpleCorpus>>
Metadata:  corpus specific: 1, document level (indexed): 0
Content:  documents: 5

[1] rt @option_snipper: $aapl beat on both eps and revenues. sees q rev. $b-$b, est. $.b http
s://t.co/hfhxqjiob
[2] rt @option_snipper: $aapl beat on both eps and revenues. sees q rev. $b-$b, est. $.b http
s://t.co/hfhxqjiob
```



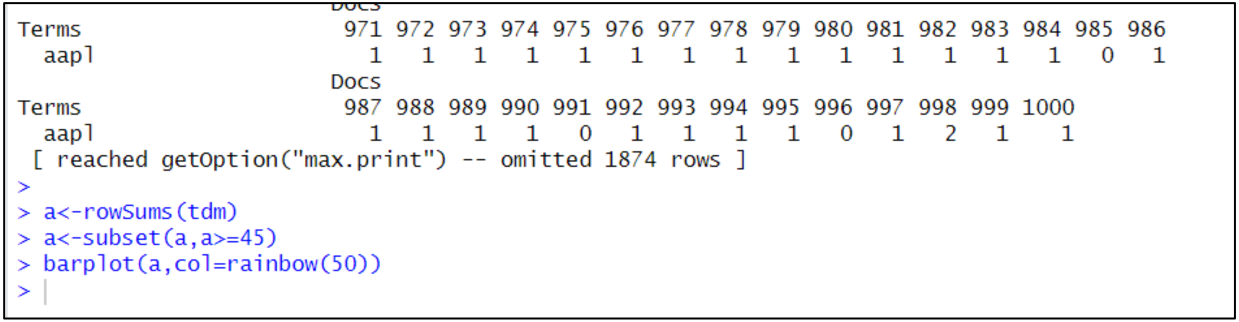
```
> corpus<-tm_map(corpus,removeWords,stopwords("english"))
Warning message:
In tm_map.SimpleCorpus(corpus, removeWords, stopwords("english")) :
  transformation drops documents
> inspect(corpus[1:5])
<<SimpleCorpus>>
Metadata:  corpus specific: 1, document level (indexed): 0
Content:  documents: 5

[1] rt @option_snipper: $aapl beat   eps  revenues. sees q rev. $b-$b, est. $.b https://t.co/h
fhxqjiob
[2] rt @option_snipper: $aapl beat   eps  revenues. sees q rev. $b-$b, est. $.b https://t.co/h
fhxqjiob
```



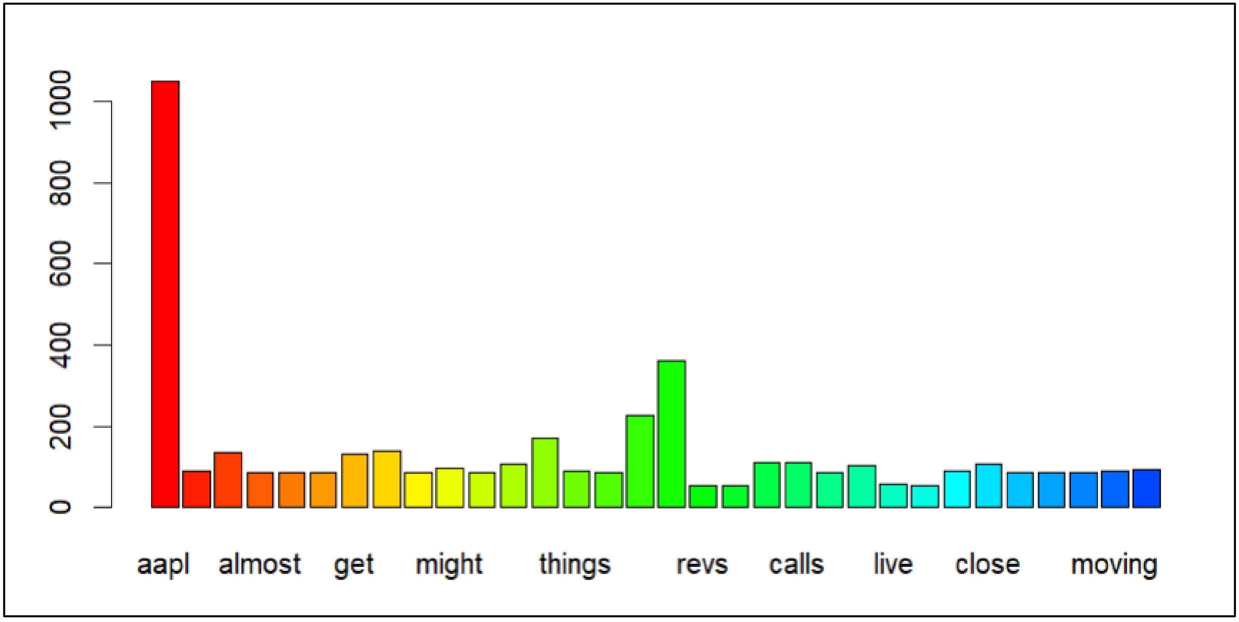
```
> corpus<-tm_map(corpus,removePunctuation)
Warning message:
In tm_map.SimpleCorpus(corpus, removePunctuation) :
  transformation drops documents
> inspect(corpus[1:5])
<<SimpleCorpus>>
Metadata:  corpus specific: 1, document level (indexed): 0
Content:  documents: 5

[1] rt optionsnipper aapl beat   eps  revenues sees q rev bb est b httpstcohfhxqjiob
[2] rt optionsnipper aapl beat   eps  revenues sees q rev bb est b httpstcohfhxqjiob
[3]  see  break  timers aapl
[4] rt sylvacap things might get ugly  aapl  iphone delay  aapl  means almost   fang stocks
pos…
[5] aapl  wow   supposed   throwaway quarter  aapl beats    million  revenue trillion dollar
company
>
```



```
> tdm<- TermDocumentMatrix(corpus)
> tdm<- as.matrix(tdm)
> tdm[1:10,1:20]
                   Docs
Terms               1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
  aapl              1 1 1 2 2 1 1 1 1  1  1  1  1  1  2  1  1  1  1  1
  beat              1 1 0 0 0 0 0 0 0  0  0  0  0  0  0  0  0  0  1  0
  eps               1 1 0 0 0 1 1 1 0  2  0  0  0  1  0  0  1  1  0  0
  est               1 1 0 0 0 2 2 0 0  0  0  0  0  2  0  0  2  2  0  0
  httpstcohfhxqjiob 1 1 0 0 0 0 0 0 0  0  0  0  0  0  0  0  0  0  0  0
  optionsnipper     1 1 0 0 0 0 0 0 0  0  0  0  0  0  0  0  0  0  0  0
  rev               1 1 0 0 0 0 0 0 0  1  0  0  1  0  0  0  0  0  0  0
  revenues          1 1 0 0 0 0 0 0 0  0  0  0  0  0  0  0  0  0  0  0
  sees              1 1 0 0 0 0 0 0 0  0  0  0  0  0  0  0  0  0  0  0
  break             0 0 1 0 0 0 0 0 0  0  0  0  0  0  0  0  0  0  0  0
> print(tdm)
                        Docs
Terms                    1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
  aapl                   1 1 1 2 2 1 1 1 1  1  1  1  1  1  2  1  1  1  1  1  1  1  1  2
                        Docs
Terms                    25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45
  aapl                    1  1  1  1  2  1  2  1  1  1  1  1  1  1  1  1  1  1  1  1  1
```

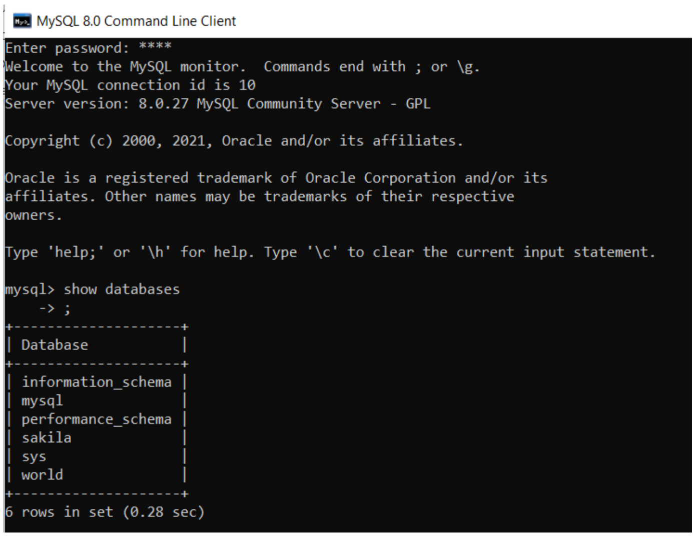


```
                        Docs
Terms                    971 972 973 974 975 976 977 978 979 980 981 982 983 984 985 986
  aapl                     1   1   1   1   1   1   1   1   1   1   1   1   1   1   0   1
                        Docs
Terms                    987 988 989 990 991 992 993 994 995 996 997 998 999 1000
  aapl                     1   1   1   1   0   1   1   1   1   0   1   2   1    1
 [ reached getOption("max.print") -- omitted 1874 rows ]
>
> a<-rowSums(tdm)
> a<-subset(a,a>=45)
> barplot(a,col=rainbow(50))
> 
```

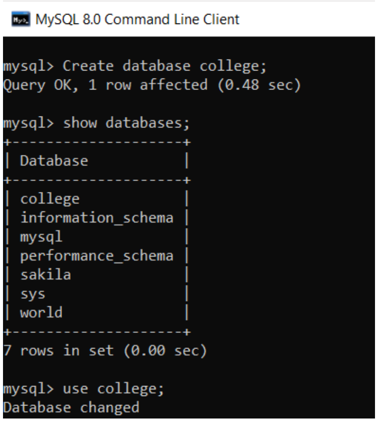


Practical 08

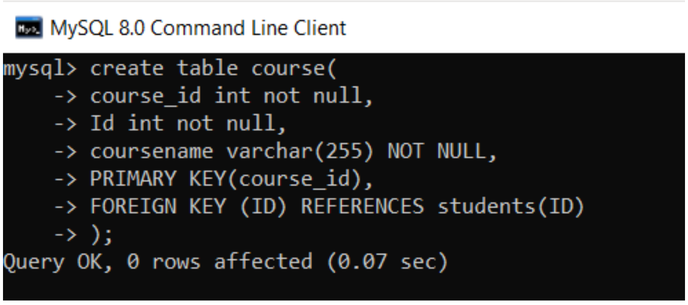
Aim: Database - Create two tables and make primary and foreign key and use select and where clause.



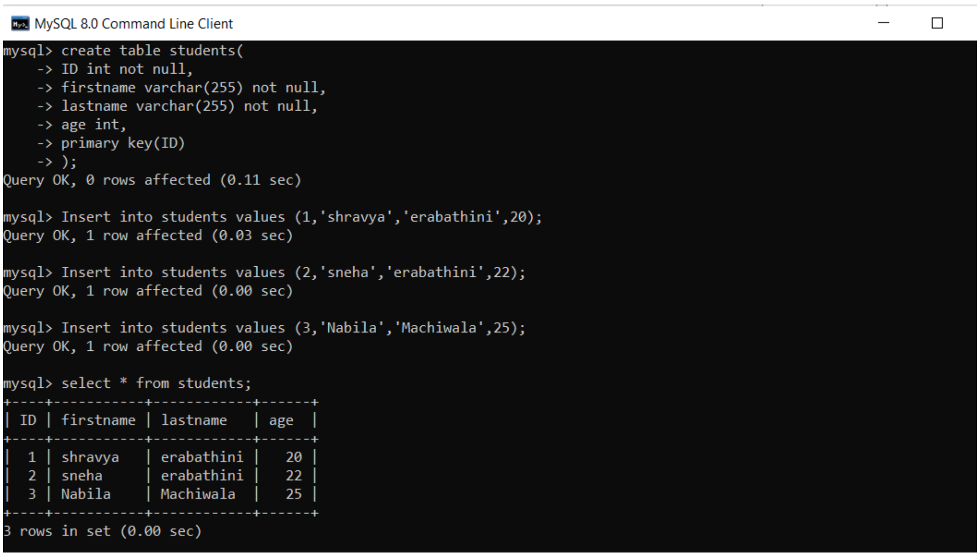
```
Enter password: ****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 10
Server version: 8.0.27 MySQL Community Server - GPL

Copyright (c) 2000, 2021, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> show databases
    -> ;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| sakila             |
| sys                |
| world              |
+--------------------+
6 rows in set (0.28 sec)
```



```
mysql> Create database college;
Query OK, 1 row affected (0.48 sec)

mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| college            |
| information_schema |
| mysql              |
| performance_schema |
| sakila             |
| sys                |
| world              |
+--------------------+
7 rows in set (0.00 sec)

mysql> use college;
Database changed
```

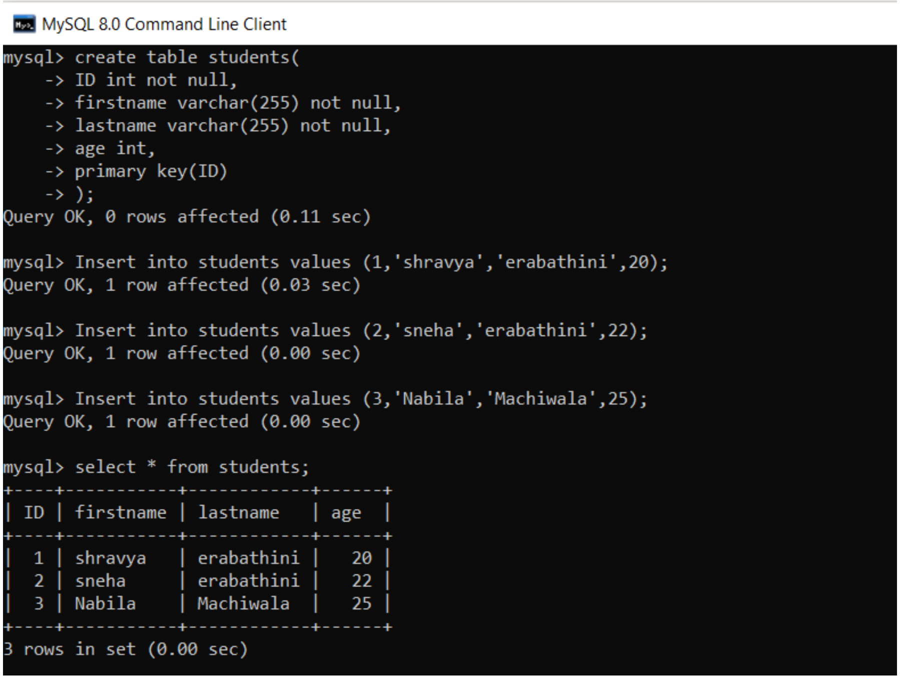


```
mysql> create table course(
    -> course_id int not null,
    -> Id int not null,
    -> coursename varchar(255) NOT NULL,
    -> PRIMARY KEY(course_id),
    -> FOREIGN KEY (ID) REFERENCES students(ID)
    -> );
Query OK, 0 rows affected (0.07 sec)
```

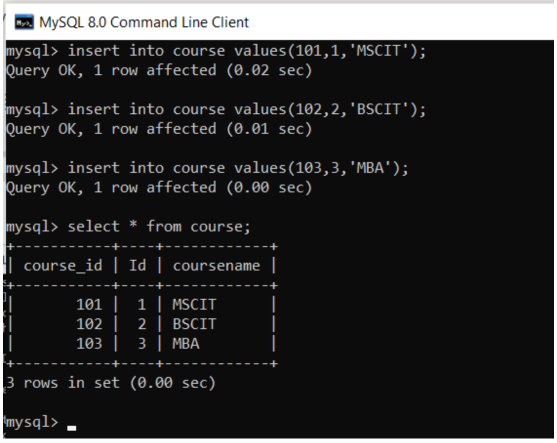


```
mysql> create table students(
    -> ID int not null,
    -> firstname varchar(255) not null,
    -> lastname varchar(255) not null,
    -> age int,
    -> primary key(ID)
    -> );
Query OK, 0 rows affected (0.11 sec)

mysql> Insert into students values (1,'shravya','erabathini',20);
Query OK, 1 row affected (0.03 sec)

mysql> Insert into students values (2,'sneha','erabathini',22);
Query OK, 1 row affected (0.00 sec)

mysql> Insert into students values (3,'Nabila','Machiwala',25);
Query OK, 1 row affected (0.00 sec)

mysql> select * from students;
+----+-----------+------------+------+
| ID | firstname | lastname   | age  |
+----+-----------+------------+------+
|  1 | shravya   | erabathini |   20 |
|  2 | sneha     | erabathini |   22 |
|  3 | Nabila    | Machiwala  |   25 |
+----+-----------+------------+------+
3 rows in set (0.00 sec)
```

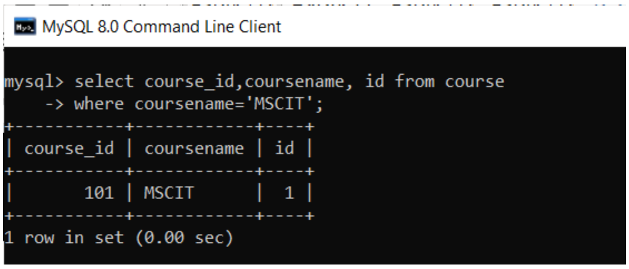
Signature



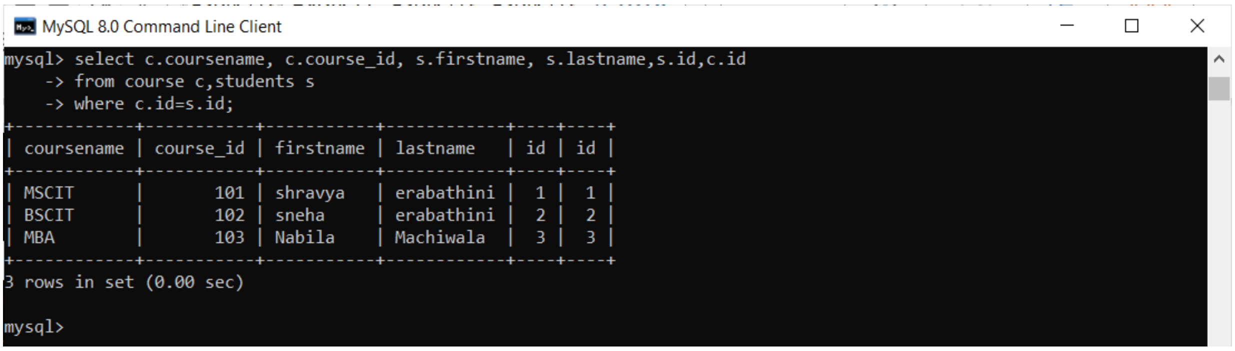
```
mysql> create table students(
    -> ID int not null,
    -> firstname varchar(255) not null,
    -> lastname varchar(255) not null,
    -> age int,
    -> primary key(ID)
    -> );
Query OK, 0 rows affected (0.11 sec)

mysql> Insert into students values (1,'shravya','erabathini',20);
Query OK, 1 row affected (0.03 sec)

mysql> Insert into students values (2,'sneha','erabathini',22);
Query OK, 1 row affected (0.00 sec)

mysql> Insert into students values (3,'Nabila','Machiwala',25);
Query OK, 1 row affected (0.00 sec)

mysql> select * from students;
+----+-----------+------------+------+
| ID | firstname | lastname   | age  |
+----+-----------+------------+------+
|  1 | shravya   | erabathini |   20 |
|  2 | sneha     | erabathini |   22 |
|  3 | Nabila    | Machiwala  |   25 |
+----+-----------+------------+------+
3 rows in set (0.00 sec)
```



```
mysql> insert into course values(101,1,'MSCIT');
Query OK, 1 row affected (0.02 sec)

mysql> insert into course values(102,2,'BSCIT');
Query OK, 1 row affected (0.01 sec)

mysql> insert into course values(103,3,'MBA');
Query OK, 1 row affected (0.00 sec)

mysql> select * from course;
+-----------+----+------------+
| course_id | Id | coursename |
+-----------+----+------------+
|       101 |  1 | MSCIT      |
|       102 |  2 | BSCIT      |
|       103 |  3 | MBA        |
+-----------+----+------------+
3 rows in set (0.00 sec)

mysql> _
```



```
mysql> select course_id,coursename, id from course
    -> where coursename='MSCIT';
+-----------+------------+----+
| course_id | coursename | id |
+-----------+------------+----+
|       101 | MSCIT      |  1 |
+-----------+------------+----+
1 row in set (0.00 sec)
```



```
mysql> select c.coursename, c.course_id, s.firstname, s.lastname,s.id,c.id
    -> from course c,students s
    -> where c.id=s.id;
+------------+-----------+-----------+------------+----+----+
| coursename | course_id | firstname | lastname   | id | id |
+------------+-----------+-----------+------------+----+----+
| MSCIT      |       101 | shravya   | erabathini |  1 |  1 |
| BSCIT      |       102 | sneha     | erabathini |  2 |  2 |
| MBA        |       103 | Nabila    | Machiwala  |  3 |  3 |
+------------+-----------+-----------+------------+----+----+
3 rows in set (0.00 sec)

mysql>
```

Signature

Practical 09

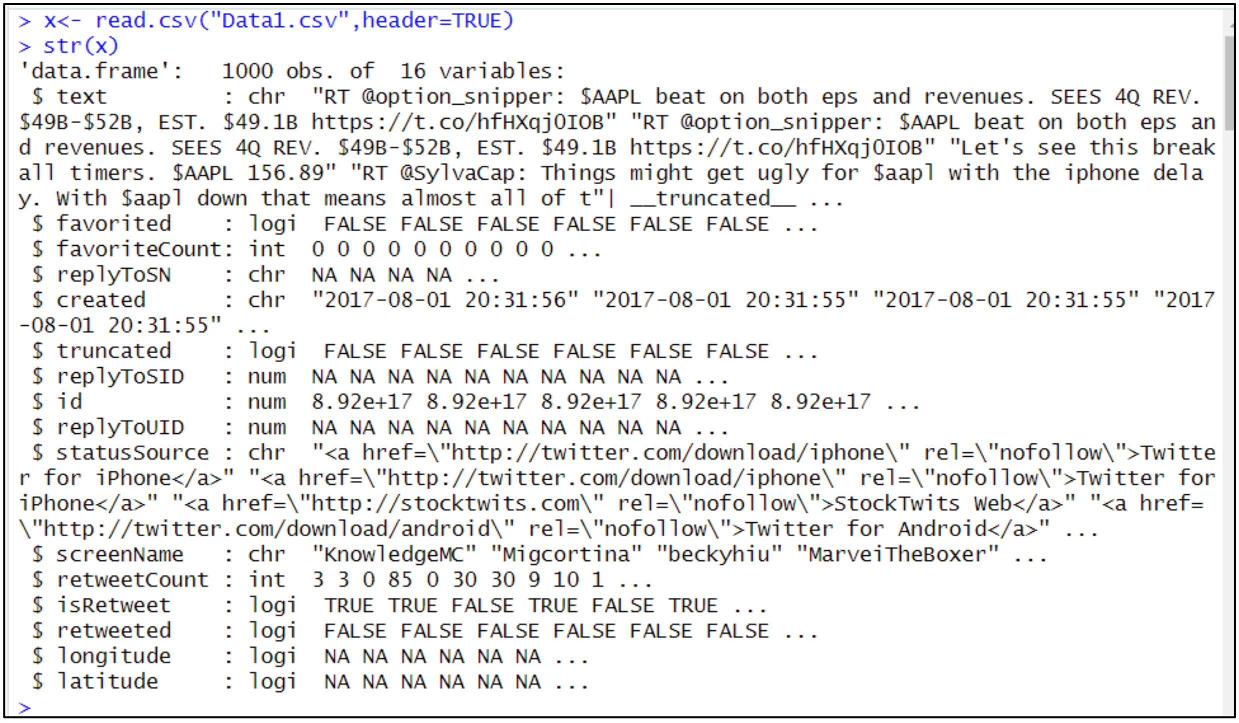
Aim: Calculate the sentiment analysis score and visualize the result.

Code:

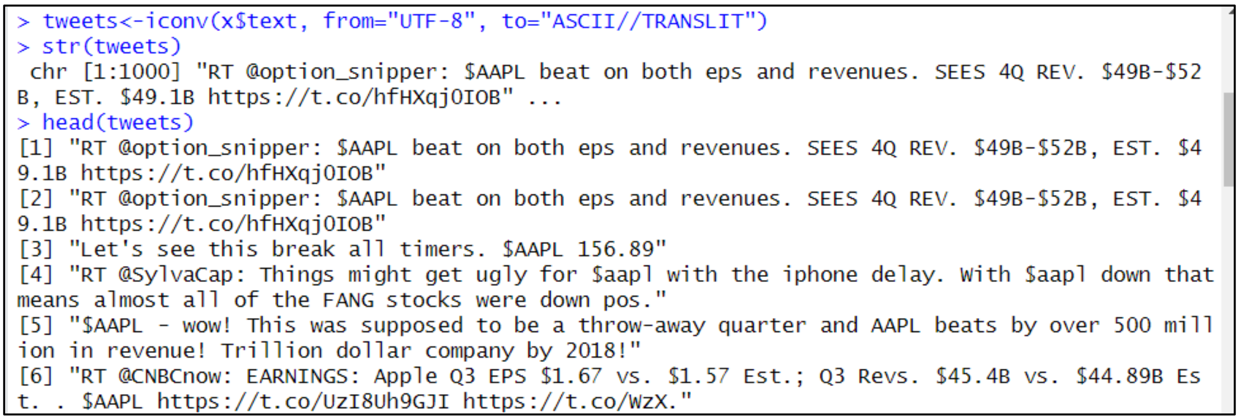
```
install.packages("rlang")
install.packages("syuzhet")
install.packages("lubridate")
install.packages("ggplot2")
install.packages("scales")
install.packages("reshape2")
install.packages("dplyr")
install.packages("tm")
install.packages("corpus")

library(rlang)
library(syuzhet) #use for sentiment analysis
library(lubridate)
library(ggplot2)
library(scales)
library(reshape2)
library(dplyr)
library(tm)
library(corpus)

x<- read.csv("Data1.csv",header=TRUE)
str(x)

tweets<-iconv(x$text, from="UTF-8", to="ASCII//TRANSLIT")
str(tweets)
head(tweets)

corpus<-iconv(x$text)
corpus<-Corpus(VectorSource(corpus))
inspect(corpus[1:5])
corpus<-tm_map(corpus,tolower)
inspect(corpus[1:5])
corpus<-tm_map(corpus,removeNumbers)
inspect(corpus[1:5])
corpus<-tm_map(corpus,removeWords,stopwords("english"))
inspect(corpus[1:5])
corpus<-tm_map(corpus,removePunctuation)
inspect(corpus[1:5])

tdm<- TermDocumentMatrix(corpus)
tdm<- as.matrix(tdm)
tdm[1:10,1:20]
print(tdm)

a<-rowSums(tdm)
a<-subset(a,a>=45)
barplot(a,col=rainbow(50))

score<-get_nrc_sentiment(tweets)
head(score)
barplot(colSums(score), las = 2, col = rainbow(10), ylab = 'Count', main = 'Sentiment Scores Tweets')
```

Signature

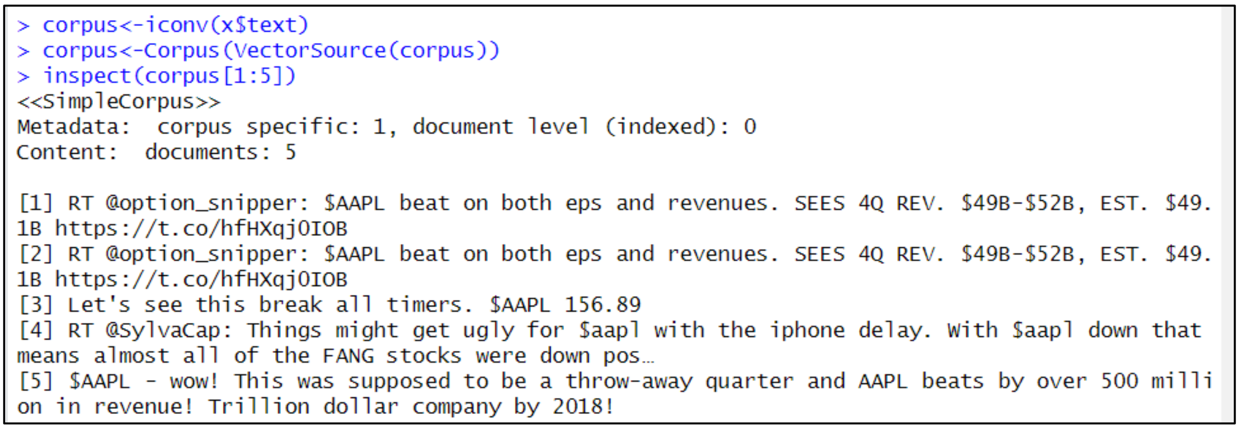
Output:



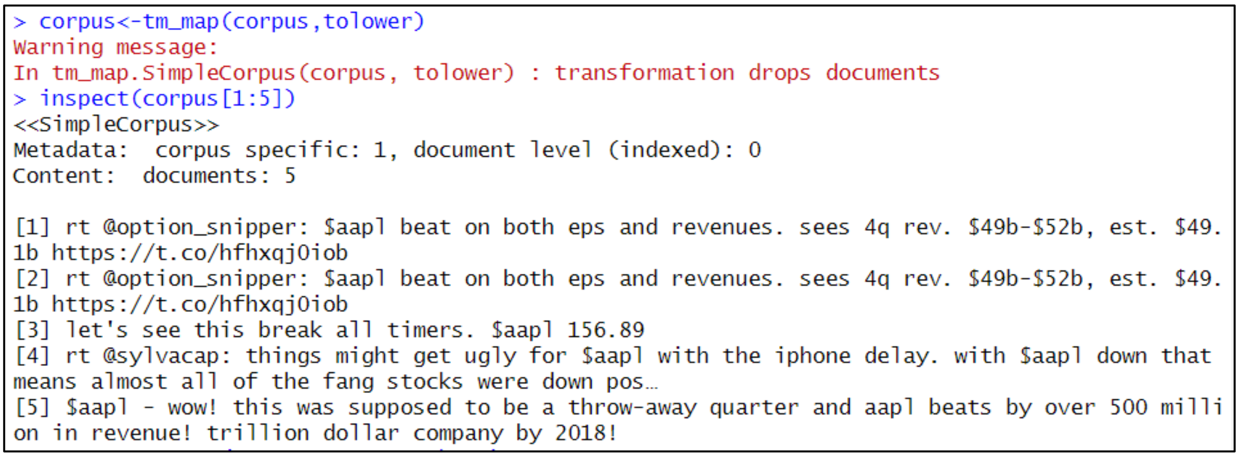
```
> x<- read.csv("Data1.csv",header=TRUE)
> str(x)
'data.frame':   1000 obs. of  16 variables:
 $ text         : chr  "RT @option_snipper: $AAPL beat on both eps and revenues. SEES 4Q REV.
$49B-$52B, EST. $49.1B https://t.co/hfHXqj0IOB" "RT @option_snipper: $AAPL beat on both eps an
d revenues. SEES 4Q REV. $49B-$52B, EST. $49.1B https://t.co/hfHXqj0IOB" "Let's see this break
all timers. $AAPL 156.89" "RT @SylvaCap: Things might get ugly for $aapl with the iphone dela
y. With $aapl down that means almost all of t"| __truncated__ ...
 $ favorited    : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
 $ favoriteCount: int  0 0 0 0 0 0 0 0 0 0 ...
 $ replyToSN    : chr  NA NA NA NA ...
 $ created      : chr  "2017-08-01 20:31:56" "2017-08-01 20:31:55" "2017-08-01 20:31:55" "2017
-08-01 20:31:55" ...
 $ truncated    : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
 $ replyToSID   : num  NA NA NA NA NA NA NA NA NA NA ...
 $ id           : num  8.92e+17 8.92e+17 8.92e+17 8.92e+17 8.92e+17 ...
 $ replyToUID   : num  NA NA NA NA NA NA NA NA NA NA ...
 $ statusSource : chr  "<a href=\"http://twitter.com/download/iphone\" rel=\"nofollow\">Twitte
r for iPhone</a>" "<a href=\"http://twitter.com/download/iphone\" rel=\"nofollow\">Twitter for
iPhone</a>" "<a href=\"http://stocktwits.com\" rel=\"nofollow\">StockTwits Web</a>" "<a href=
\"http://twitter.com/download/android\" rel=\"nofollow\">Twitter for Android</a>" ...
 $ screenName   : chr  "KnowledgeMC" "Migcortina" "beckyhiu" "MarveiTheBoxer" ...
 $ retweetCount : int  3 3 0 85 0 30 30 9 10 1 ...
 $ isRetweet    : logi  TRUE TRUE FALSE TRUE FALSE TRUE ...
 $ retweeted    : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
 $ longitude    : logi  NA NA NA NA NA NA ...
 $ latitude     : logi  NA NA NA NA NA NA ...
>
```



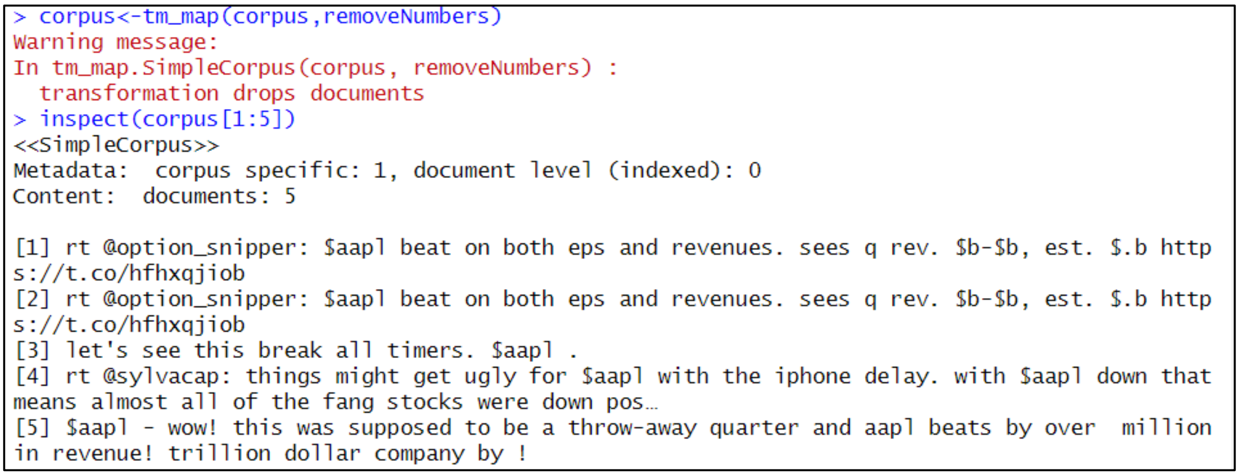
```
> tweets<-iconv(x$text, from="UTF-8", to="ASCII//TRANSLIT")
> str(tweets)
 chr [1:1000] "RT @option_snipper: $AAPL beat on both eps and revenues. SEES 4Q REV. $49B-$52
B, EST. $49.1B https://t.co/hfHXqj0IOB" ...
> head(tweets)
[1] "RT @option_snipper: $AAPL beat on both eps and revenues. SEES 4Q REV. $49B-$52B, EST. $4
9.1B https://t.co/hfHXqj0IOB"
[2] "RT @option_snipper: $AAPL beat on both eps and revenues. SEES 4Q REV. $49B-$52B, EST. $4
9.1B https://t.co/hfHXqj0IOB"
[3] "Let's see this break all timers. $AAPL 156.89"
[4] "RT @SylvaCap: Things might get ugly for $aapl with the iphone delay. With $aapl down that
means almost all of the FANG stocks were down pos."
[5] "$AAPL - wow! This was supposed to be a throw-away quarter and AAPL beats by over 500 mill
ion in revenue! Trillion dollar company by 2018!"
[6] "RT @CNBCnow: EARNINGS: Apple Q3 EPS $1.67 vs. $1.57 Est.; Q3 Revs. $45.4B vs. $44.89B Es
t. . $AAPL https://t.co/UzI8Uh9GJI https://t.co/WzX."
```



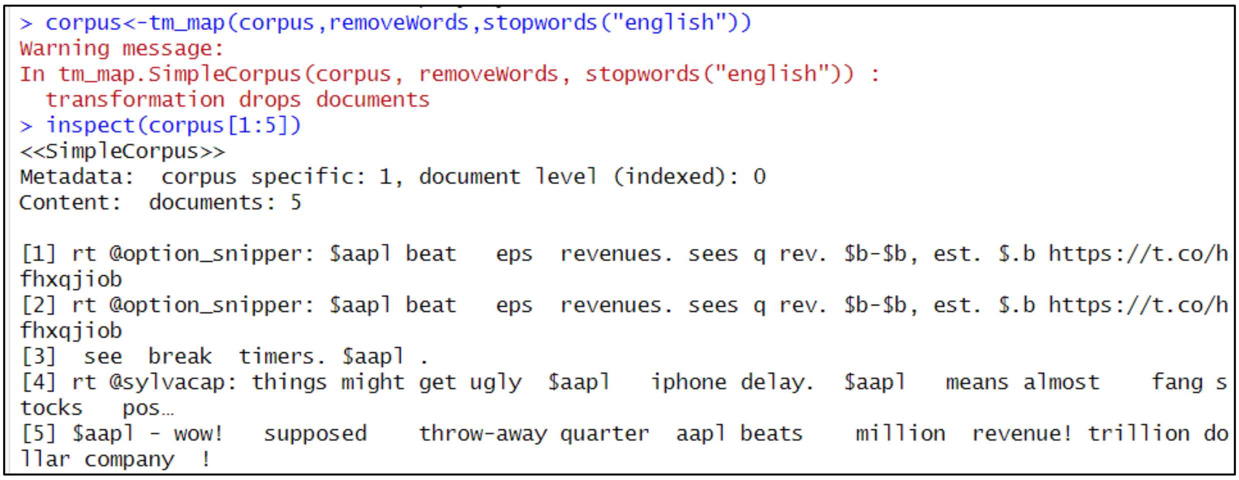
```
> corpus<-iconv(x$text)
> corpus<-Corpus(VectorSource(corpus))
> inspect(corpus[1:5])
<<SimpleCorpus>>
Metadata:  corpus specific: 1, document level (indexed): 0
Content:  documents: 5

[1] RT @option_snipper: $AAPL beat on both eps and revenues. SEES 4Q REV. $49B-$52B, EST. $49.
1B https://t.co/hfHXqj0IOB
[2] RT @option_snipper: $AAPL beat on both eps and revenues. SEES 4Q REV. $49B-$52B, EST. $49.
1B https://t.co/hfHXqj0IOB
[3] Let's see this break all timers. $AAPL 156.89
[4] RT @SylvaCap: Things might get ugly for $aapl with the iphone delay. With $aapl down that
means almost all of the FANG stocks were down pos…
[5] $AAPL - wow! This was supposed to be a throw-away quarter and AAPL beats by over 500 milli
on in revenue! Trillion dollar company by 2018!
```



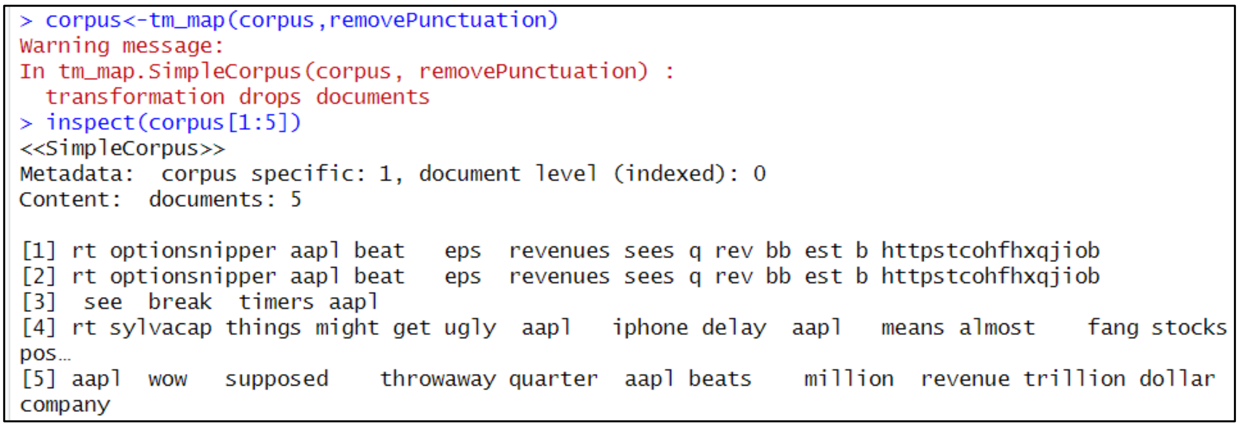
```
> corpus<-tm_map(corpus,tolower)
Warning message:
In tm_map.SimpleCorpus(corpus, tolower) : transformation drops documents
> inspect(corpus[1:5])
<<SimpleCorpus>>
Metadata:  corpus specific: 1, document level (indexed): 0
Content:  documents: 5

[1] rt @option_snipper: $aapl beat on both eps and revenues. sees 4q rev. $49b-$52b, est. $49.
1b https://t.co/hfhxqj0iob
[2] rt @option_snipper: $aapl beat on both eps and revenues. sees 4q rev. $49b-$52b, est. $49.
1b https://t.co/hfhxqj0iob
[3] let's see this break all timers. $aapl 156.89
[4] rt @sylvacap: things might get ugly for $aapl with the iphone delay. with $aapl down that
means almost all of the fang stocks were down pos…
[5] $aapl - wow! this was supposed to be a throw-away quarter and aapl beats by over 500 milli
on in revenue! trillion dollar company by 2018!
```



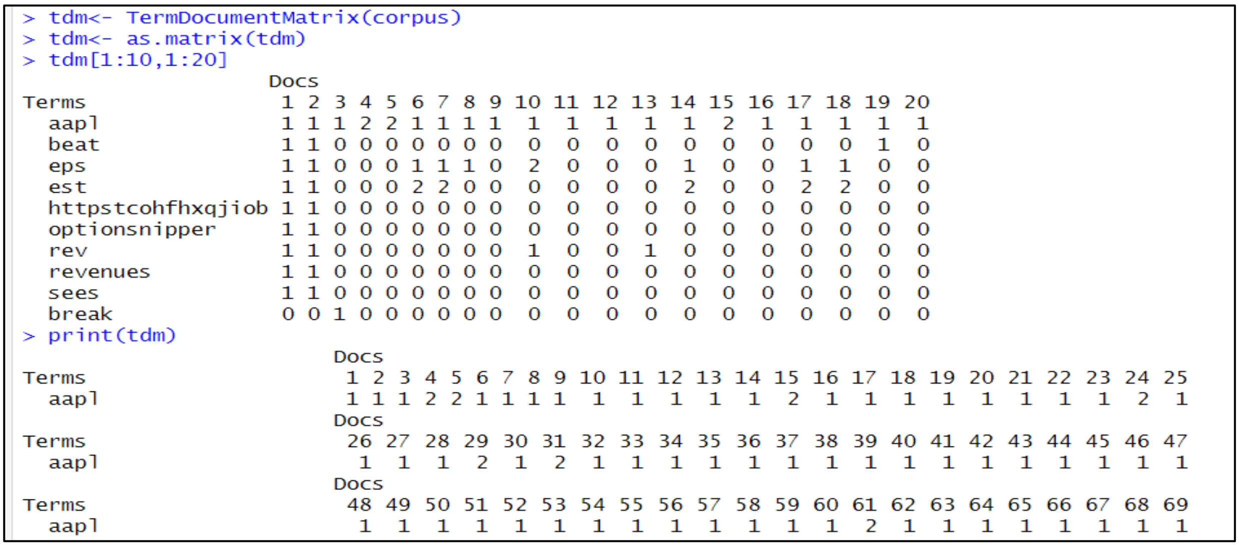
```
> corpus<-tm_map(corpus,removeNumbers)
Warning message:
In tm_map.SimpleCorpus(corpus, removeNumbers) :
  transformation drops documents
> inspect(corpus[1:5])
<<SimpleCorpus>>
Metadata:  corpus specific: 1, document level (indexed): 0
Content:  documents: 5

[1] rt @option_snipper: $aapl beat on both eps and revenues. sees q rev. $b-$b, est. $.b http
s://t.co/hfhxqjiob
[2] rt @option_snipper: $aapl beat on both eps and revenues. sees q rev. $b-$b, est. $.b http
s://t.co/hfhxqjiob
[3] let's see this break all timers. $aapl .
[4] rt @sylvacap: things might get ugly for $aapl with the iphone delay. with $aapl down that
means almost all of the fang stocks were down pos…
[5] $aapl - wow! this was supposed to be a throw-away quarter and aapl beats by over  million
in revenue! trillion dollar company by !
```



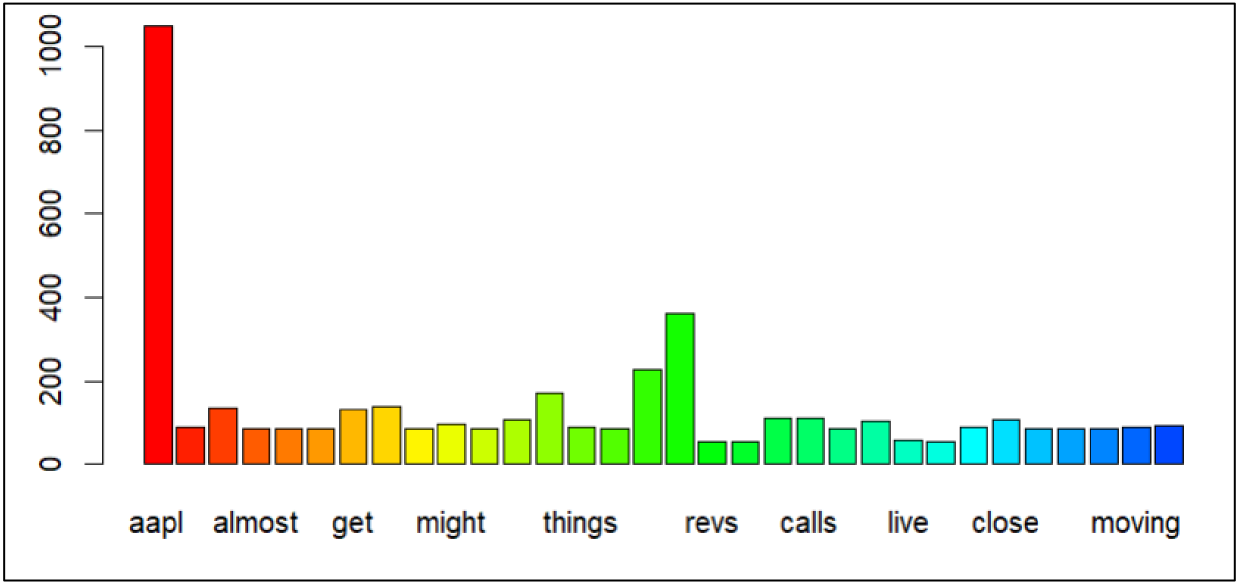
```
> corpus<-tm_map(corpus,removeWords,stopwords("english"))
Warning message:
In tm_map.SimpleCorpus(corpus, removeWords, stopwords("english")) :
  transformation drops documents
> inspect(corpus[1:5])
<<SimpleCorpus>>
Metadata:  corpus specific: 1, document level (indexed): 0
Content:  documents: 5

[1] rt @option_snipper: $aapl beat   eps   revenues. sees q rev. $b-$b, est. $.b https://t.co/h
fhxqjiob
[2] rt @option_snipper: $aapl beat   eps   revenues. sees q rev. $b-$b, est. $.b https://t.co/h
fhxqjiob
[3]  see  break  timers. $aapl .
[4] rt @sylvacap: things might get ugly  $aapl   iphone delay.  $aapl   means almost    fang s
tocks   pos…
[5] $aapl - wow!   supposed    throw-away quarter  aapl beats    million  revenue! trillion do
llar company  !
```

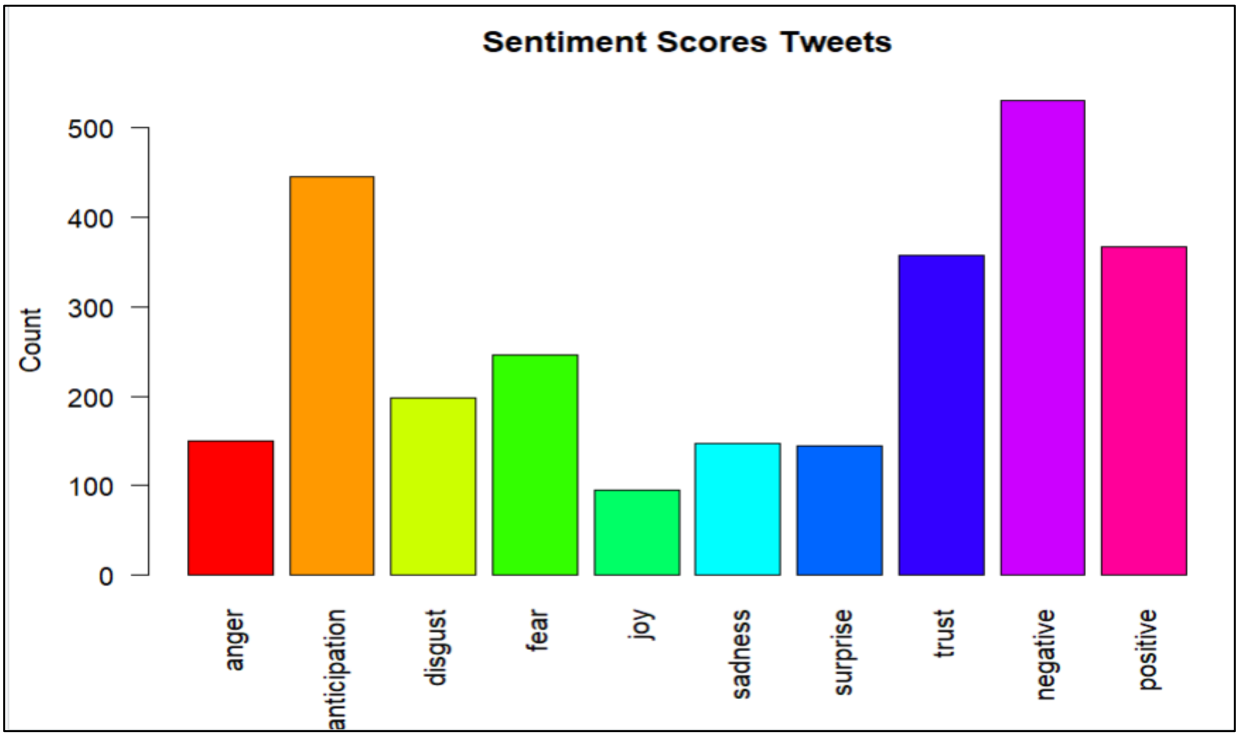


```
> corpus<-tm_map(corpus,removePunctuation)
Warning message:
In tm_map.SimpleCorpus(corpus, removePunctuation) :
  transformation drops documents
> inspect(corpus[1:5])
<<SimpleCorpus>>
Metadata:  corpus specific: 1, document level (indexed): 0
Content:  documents: 5

[1] rt optionsnipper aapl beat   eps   revenues sees q rev bb est b httpstcohfhxqjiob
[2] rt optionsnipper aapl beat   eps   revenues sees q rev bb est b httpstcohfhxqjiob
[3]  see  break  timers aapl
[4] rt sylvacap things might get ugly  aapl   iphone delay  aapl   means almost    fang stocks
pos…
[5] aapl  wow   supposed    throwaway quarter  aapl beats    million  revenue trillion dollar
company
```



```
> tdm<- TermDocumentMatrix(corpus)
> tdm<- as.matrix(tdm)
> tdm[1:10,1:20]
                   Docs
Terms               1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
  aapl              1 1 1 2 2 1 1 1 1  1  1  1  1  1  2  1  1  1  1  1
  beat              1 1 0 0 0 0 0 0 0  0  0  0  0  0  0  0  0  0  1  0
  eps               1 1 0 0 0 1 1 1 0  2  0  0  0  1  0  0  1  1  0  0
  est               1 1 0 0 0 2 2 0 0  0  0  0  0  2  0  0  2  2  0  0
  httpstcohfhxqjiob 1 1 0 0 0 0 0 0 0  0  0  0  0  0  0  0  0  0  0  0
  optionsnipper     1 1 0 0 0 0 0 0 0  0  0  0  0  0  0  0  0  0  0  0
  rev               1 1 0 0 0 0 0 0 0  1  0  0  1  0  0  0  0  0  0  0
  revenues          1 1 0 0 0 0 0 0 0  0  0  0  0  0  0  0  0  0  0  0
  sees              1 1 0 0 0 0 0 0 0  0  0  0  0  0  0  0  0  0  0  0
  break             0 0 1 0 0 0 0 0 0  0  0  0  0  0  0  0  0  0  0  0
> print(tdm)
                        Docs
Terms                    1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25
  aapl                   1 1 1 2 2 1 1 1 1  1  1  1  1  1  2  1  1  1  1  1  1  1  1  2  1
                        Docs
Terms                    26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47
  aapl                    1  1  1  2  1  2  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
                        Docs
Terms                    48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69
  aapl                    1  1  1  1  1  1  1  1  1  1  1  1  1  2  1  1  1  1  1  1  1  1
```

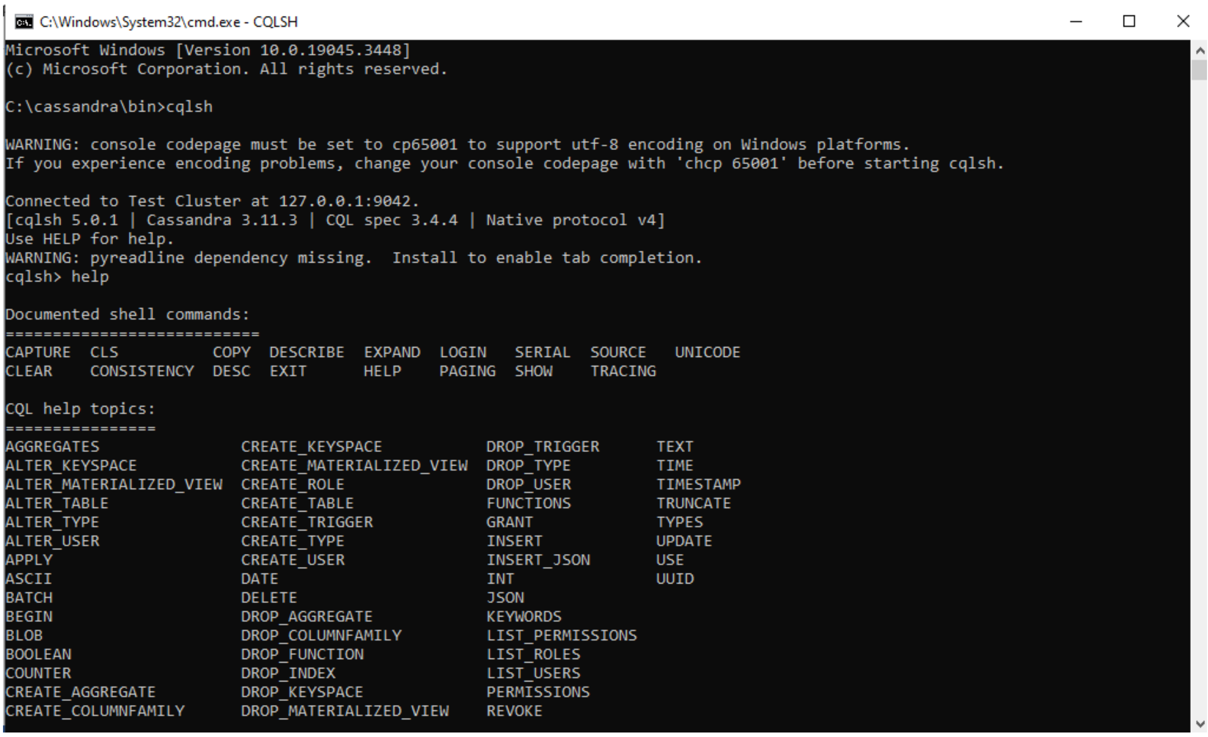




Practical 10A

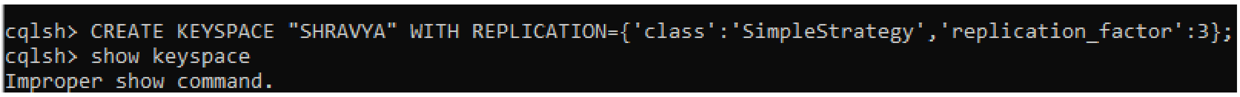
Aim: Creating Data Model using Cassandra

- cqlsh
- help



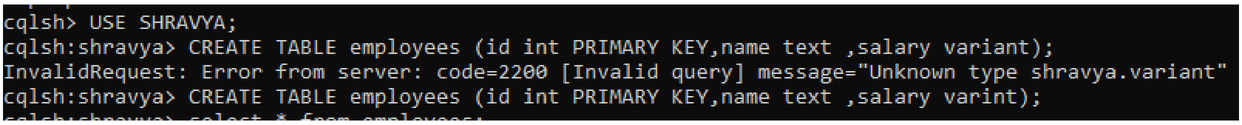
```
Microsoft Windows [Version 10.0.19045.3448]
(c) Microsoft Corporation. All rights reserved.

C:\cassandra\bin>cqlsh

WARNING: console codepage must be set to cp65001 to support utf-8 encoding on Windows platforms.
If you experience encoding problems, change your console codepage with 'chcp 65001' before starting cqlsh.

Connected to Test Cluster at 127.0.0.1:9042.
[cqlsh 5.0.1 | Cassandra 3.11.3 | CQL spec 3.4.4 | Native protocol v4]
Use HELP for help.
WARNING: pyreadline dependency missing.  Install to enable tab completion.
cqlsh> help

Documented shell commands:
===========================
CAPTURE  CLS          COPY  DESCRIBE  EXPAND  LOGIN   SERIAL  SOURCE   UNICODE
CLEAR    CONSISTENCY  DESC  EXIT      HELP    PAGING  SHOW    TRACING

CQL help topics:
================
AGGREGATES               CREATE_KEYSPACE           DROP_TRIGGER      TEXT
ALTER_KEYSPACE           CREATE_MATERIALIZED_VIEW  DROP_TYPE         TIME
ALTER_MATERIALIZED_VIEW  CREATE_ROLE               DROP_USER         TIMESTAMP
ALTER_TABLE              CREATE_TABLE              FUNCTIONS         TRUNCATE
ALTER_TYPE               CREATE_TRIGGER            GRANT             TYPES
ALTER_USER               CREATE_TYPE               INSERT            UPDATE
APPLY                    CREATE_USER               INSERT_JSON       USE
ASCII                    DATE                      INT               UUID
BATCH                    DELETE                    JSON
BEGIN                    DROP_AGGREGATE            KEYWORDS
BLOB                     DROP_COLUMNFAMILY         LIST_PERMISSIONS
BOOLEAN                  DROP_FUNCTION             LIST_ROLES
COUNTER                  DROP_INDEX                LIST_USERS
CREATE_AGGREGATE         DROP_KEYSPACE             PERMISSIONS
CREATE_COLUMNFAMILY      DROP_MATERIALIZED_VIEW    REVOKE
```

- Create Keyspace Shravya with REPLICATION = {‘class’:’SimpleStrategy’,’replication_factor’:3);
- Show keyspace;



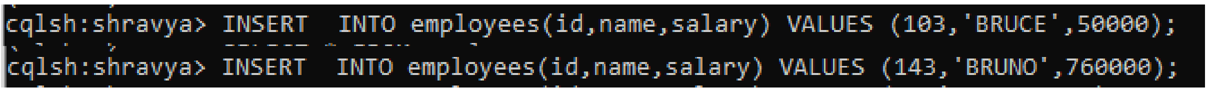
```
cqlsh> CREATE KEYSPACE "SHRAVYA" WITH REPLICATION={'class':'SimpleStrategy','replication_factor':3};
cqlsh> show keyspace
Improper show command.
```

- Use shravya;
- Create table employees(id int PRIMARY KEY,name text,salary varint);

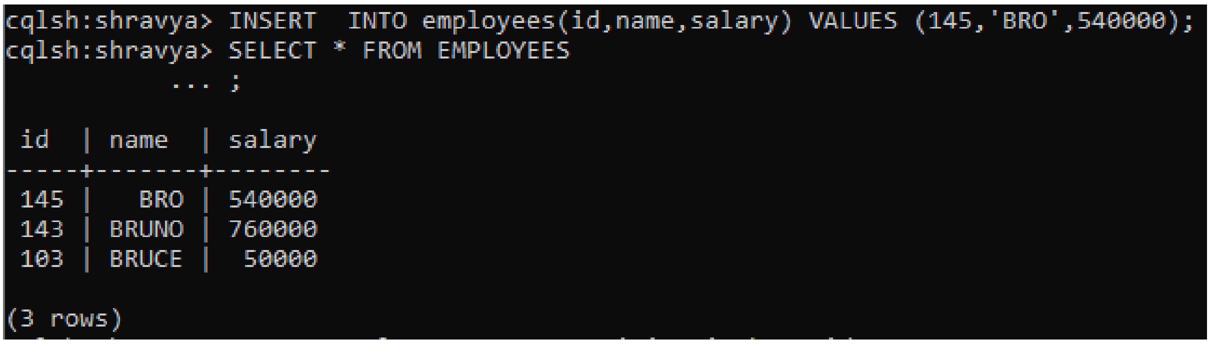


```
cqlsh> USE SHRAVYA;
cqlsh:shravya> CREATE TABLE employees (id int PRIMARY KEY,name text ,salary variant);
InvalidRequest: Error from server: code=2200 [Invalid query] message="Unknown type shravya.variant"
cqlsh:shravya> CREATE TABLE employees (id int PRIMARY KEY,name text ,salary varint);
```

- Insert into employees(id,name,salary) values(103,’BRUNCE’,50000);
- Select * from employees

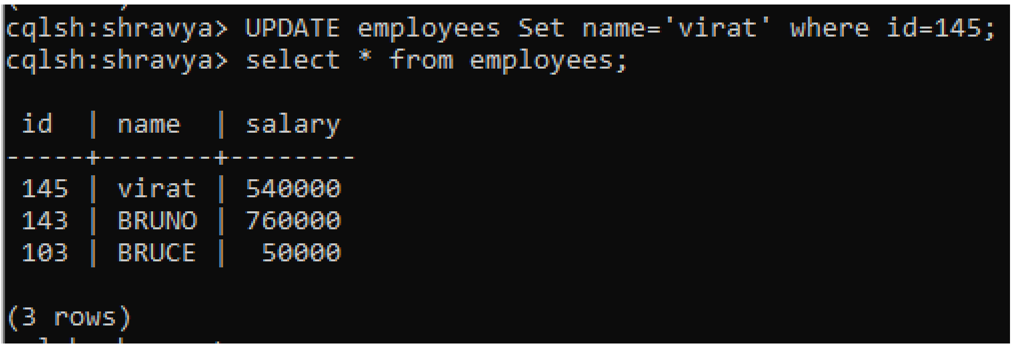


```
cqlsh:shravya> INSERT  INTO employees(id,name,salary) VALUES (103,'BRUCE',50000);
cqlsh:shravya> INSERT  INTO employees(id,name,salary) VALUES (143,'BRUNO',760000);
```



```
cqlsh:shravya> INSERT  INTO employees(id,name,salary) VALUES (145,'BRO',540000);
cqlsh:shravya> SELECT * FROM EMPLOYEES
           ... ;

 id  | name  | salary
-----+-------+--------
 145 |   BRO | 540000
 143 | BRUNO | 760000
 103 | BRUCE |  50000

(3 rows)
```

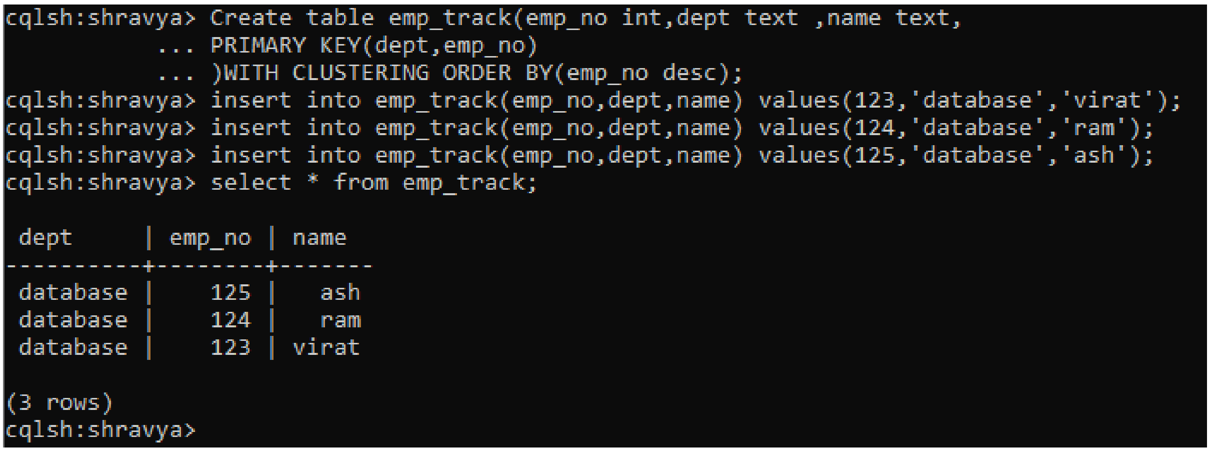
- Update employees set name='virat' where id=145;



```
cqlsh:shravya> UPDATE employees Set name='virat' where id=145;
cqlsh:shravya> select * from employees;

 id  | name  | salary
-----+-------+--------
 145 | virat | 540000
 143 | BRUNO | 760000
 103 | BRUCE |  50000

(3 rows)
```

Create clustering on emp_no in the descending order

- Create table emp_track(emp_no int,dept text,name text,PRIMARY KEY(dept,emp_no)) WITH Clustering ORDER BY(emp_no desc);
- Insert into emp_track(emp_no,dept,name) values(123,'database','virat');
- Select * from emp_track;



```
cqlsh:shravya> Create table emp_track(emp_no int,dept text ,name text,
           ... PRIMARY KEY(dept,emp_no)
           ... )WITH CLUSTERING ORDER BY(emp_no desc);
cqlsh:shravya> insert into emp_track(emp_no,dept,name) values(123,'database','virat');
cqlsh:shravya> insert into emp_track(emp_no,dept,name) values(124,'database','ram');
cqlsh:shravya> insert into emp_track(emp_no,dept,name) values(125,'database','ash');
cqlsh:shravya> select * from emp_track;

 dept     | emp_no | name
----------+--------+-------
 database |    125 |   ash
 database |    124 |   ram
 database |    123 | virat

(3 rows)
cqlsh:shravya>
```

Practical 10B

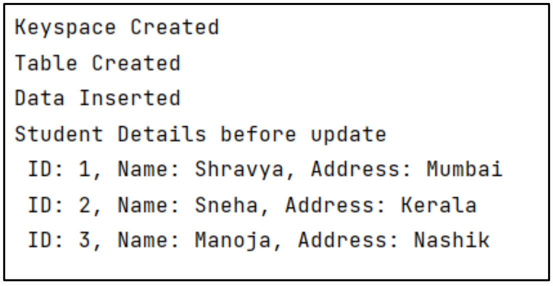
Aim: Create, Insert, Update and display the data from cassandra using python

Code:

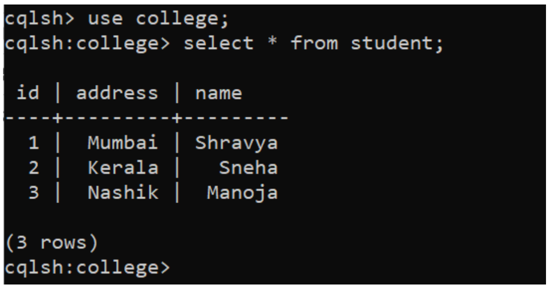
```
from cassandra.cluster import Cluster

# Connect to Cassandra
cluster = Cluster(['127.0.0.1'])
session = cluster.connect()

keyspace_name = 'College'
replication_options = {
    'class': 'SimpleStrategy',
    'replication_factor': 1
}

create_keyspace_query = f"""
    CREATE KEYSPACE IF NOT EXISTS {keyspace_name}
    WITH REPLICATION = {str(replication_options)}
"""

session.execute(create_keyspace_query)
print("Keyspace Created")

create_query=session.prepare("CREATE TABLE College.Student (id int PRIMARY KEY, name text, address text)")
session.execute(create_query)
print("Table Created")

session.execute("INSERT INTO College.Student (id, name, address) VALUES (1, 'Shravya', 'Mumbai')")
session.execute("INSERT INTO College.Student (id, name, address) VALUES (2, 'Sneha', 'Kerala')")
session.execute("INSERT INTO College.Student (id, name, address) VALUES (3, 'Manoja', 'Nashik')")
print("Data Inserted")
select_query = "SELECT * FROM College.Student"

result = session.execute(select_query)
print("Student Details before update")
for row in result:
    print(f" ID: {row.id}, Name: {row.name}, Address: {row.address}")


update_query = session.prepare("UPDATE College.Student SET address = 'Delhi' WHERE id = 2")
session.execute(update_query)
print("Data Updated")
print("Data after update")

select_query1 = "SELECT * FROM College.Student"

result = session.execute(select_query1)
print("Student Details")
for row in result:
    print(f" ID: {row.id}, Name: {row.name}, Address: {row.address}")
session.shutdown()
cluster.shutdown()
```

Signature

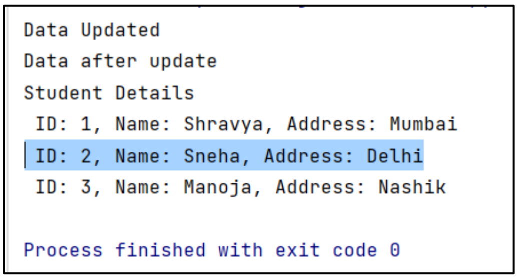
Output:



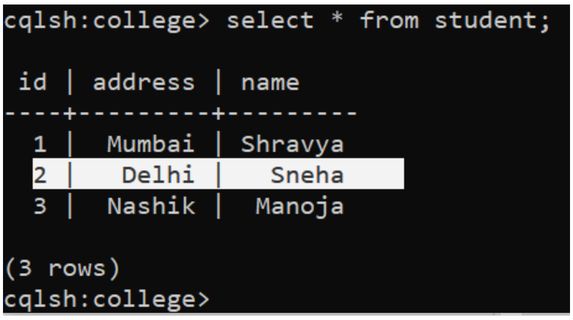
```
Keyspace Created
Table Created
Data Inserted
Student Details before update
 ID: 1, Name: Shravya, Address: Mumbai
 ID: 2, Name: Sneha, Address: Kerala
 ID: 3, Name: Manoja, Address: Nashik
```



```
cqlsh> use college;
cqlsh:college> select * from student;

 id | address | name
----+---------+---------
  1 |  Mumbai | Shravya
  2 |  Kerala |    Sneha
  3 |  Nashik |  Manoja

(3 rows)
cqlsh:college>
```



```
Data Updated
Data after update
Student Details
 ID: 1, Name: Shravya, Address: Mumbai
 ID: 2, Name: Sneha, Address: Delhi
 ID: 3, Name: Manoja, Address: Nashik

Process finished with exit code 0
```



```
cqlsh:college> select * from student;

 id | address | name
----+---------+---------
  1 |  Mumbai | Shravya
  2 |   Delhi |    Sneha
  3 |  Nashik |  Manoja

(3 rows)
cqlsh:college>
```