Assignment 3
FIT1043
Kang Hong Bo
32684673

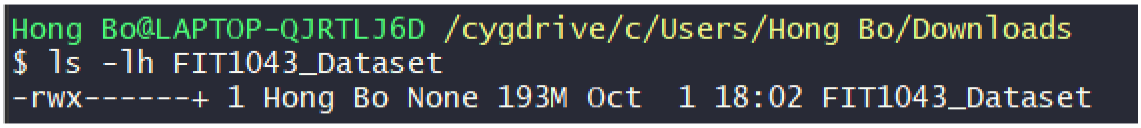
Part A

Before starting the assignment we will need to unzip the file we can unzip it by using the bash shell with the code $ gunzip FIT1043_Dataset.gz

1.

Ans: 193Mb

Code: $ ls -lh FIT1043_Dataset



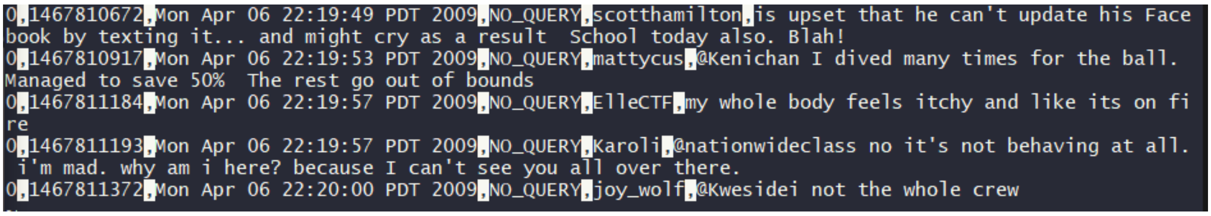
```
Hong Bo@LAPTOP-QJRTLJ6D /cygdrive/c/Users/Hong Bo/Downloads
$ ls -lh FIT1043_Dataset
-rwx------+ 1 Hong Bo None 193M Oct  1 18:02 FIT1043_Dataset
```

2.

Ans: The comma sign could be used as the separator of columns. Because the element in between the comma is similar to other lines so that we can separate the columns by using the comma sign.

Code:$ head -5 FIT1043_Dataset | less

Then /, to highlight.

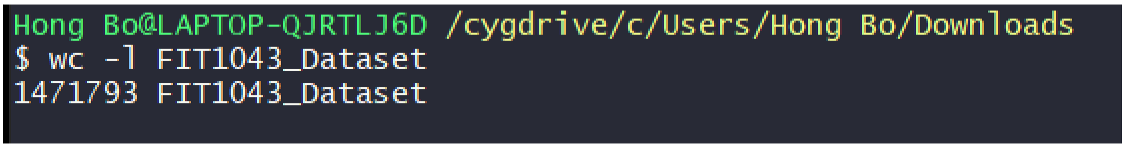


```
0,1467810672,Mon Apr 06 22:19:49 PDT 2009,NO_QUERY,scotthamilton,is upset that he can't update his Face
book by texting it... and might cry as a result  School today also. Blah!
0,1467810917,Mon Apr 06 22:19:53 PDT 2009,NO_QUERY,mattycus,@Kenichan I dived many times for the ball.
Managed to save 50%  The rest go out of bounds
0,1467811184,Mon Apr 06 22:19:57 PDT 2009,NO_QUERY,ElleCTF,my whole body feels itchy and like its on fi
re
0,1467811193,Mon Apr 06 22:19:57 PDT 2009,NO_QUERY,Karoli,@nationwideclass no it's not behaving at all.
 i'm mad. why am i here? because I can't see you all over there.
0,1467811372,Mon Apr 06 22:20:00 PDT 2009,NO_QUERY,joy_wolf,@Kwesidei not the whole crew
```

3.

Ans: 1471793 lines in the file

Code:$ wc -l FIT1043_Dataset

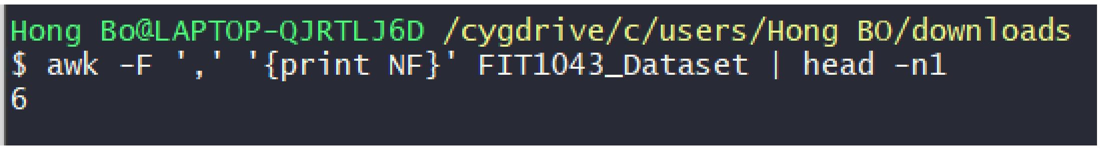


```
Hong Bo@LAPTOP-QJRTLJ6D /cygdrive/c/Users/Hong Bo/Downloads
$ wc -l FIT1043_Dataset
1471793 FIT1043_Dataset
```

4.

Ans: 6 columns are there in this file. Print NF is getting the number of fields in the current records.

Code: $ awk -F ',' '{print NF}' FIT1043_Dataset |head -n1

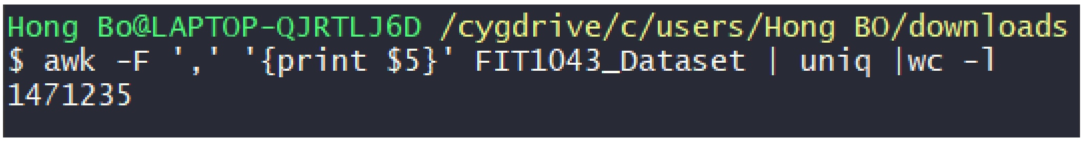


```
Hong Bo@LAPTOP-QJRTLJ6D /cygdrive/c/users/Hong BO/downloads
$ awk -F ',' '{print NF}' FIT1043_Dataset | head -n1
6
```

5.

Ans: The command uniq will get the name in lines then we count the number of line then we will get the unique name.

Code: $ awk -F ',' '{print $5}' FIT1043_Dataset | uniq | wc -l



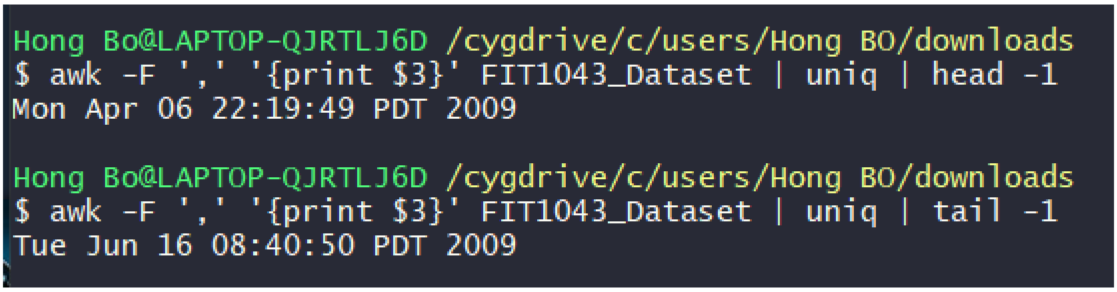
```
Hong Bo@LAPTOP-QJRTLJ6D /cygdrive/c/users/Hong BO/downloads
$ awk -F ',' '{print $5}' FIT1043_Dataset | uniq |wc -l
1471235
```

6.

Ans: The date range for the twitter posts in this file is 6-4-2009 ~ 16-6-2009.

Code: $ awk -F ',' '{print $3}' FIT1043_Dataset |uniq|head -1

$ awk -F ',' '{print $3}' FIT1043_Dataset |uniq|tail -1

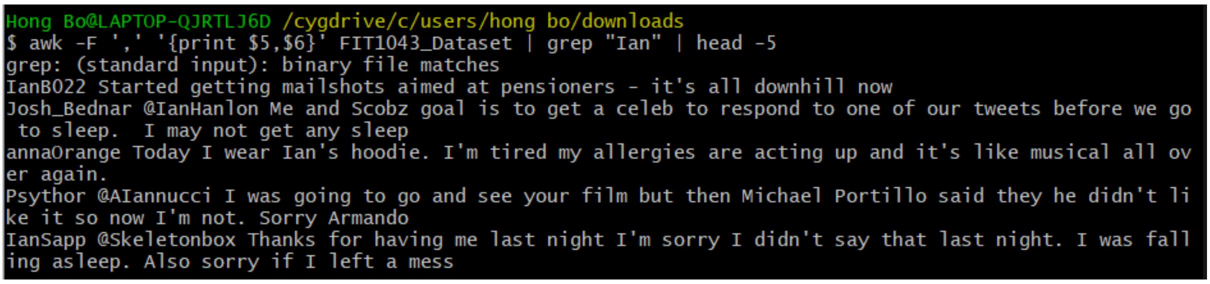


```
Hong Bo@LAPTOP-QJRTLJ6D /cygdrive/c/users/Hong BO/downloads
$ awk -F ',' '{print $3}' FIT1043_Dataset | uniq | head -1
Mon Apr 06 22:19:49 PDT 2009

Hong Bo@LAPTOP-QJRTLJ6D /cygdrive/c/users/Hong BO/downloads
$ awk -F ',' '{print $3}' FIT1043_Dataset | uniq | tail -1
Tue Jun 16 08:40:50 PDT 2009
```

7.

Ans: As we can see the first 5 grep Ian we can see that the first who mention Ian is annaOrange which is the users.

Code: $ awk -F ',' '{print $5$6}' FIT1043_Dataset |grep "Ian" |head -5

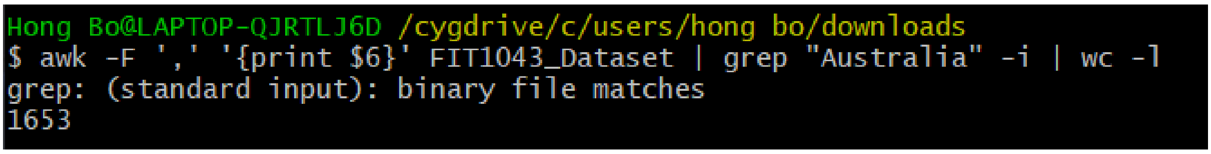


```
Hong Bo@LAPTOP-QJRTLJ6D /cygdrive/c/users/hong bo/downloads
$ awk -F ',' '{print $5,$6}' FIT1043_Dataset | grep "Ian" | head -5
grep: (standard input): binary file matches
IanB022 Started getting mailshots aimed at pensioners - it's all downhill now
Josh_Bednar @IanHanlon Me and Scobz goal is to get a celeb to respond to one of our tweets before we go
 to sleep.  I may not get any sleep
annaOrange Today I wear Ian's hoodie. I'm tired my allergies are acting up and it's like musical all ov
er again.
Psythor @AIannucci I was going to go and see your film but then Michael Portillo said they he didn't li
ke it so now I'm not. Sorry Armando
IanSapp @Skeletonbox Thanks for having me last night I'm sorry I didn't say that last night. I was fall
ing asleep. Also sorry if I left a mess
```

8.

Ans:1653 tweets with the word “Australia” in the message columns.

Code: $ awk -F ',' '{print $6}' FIT1043_Dataset | grep "Australia" -i | wc -l

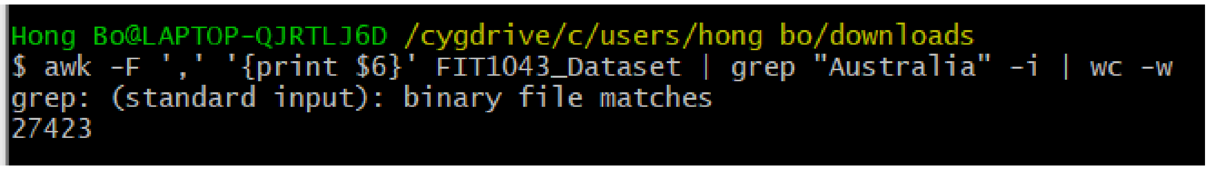


```
Hong Bo@LAPTOP-QJRTLJ6D /cygdrive/c/users/hong bo/downloads
$ awk -F ',' '{print $6}' FIT1043_Dataset | grep "Australia" -i | wc -l
grep: (standard input): binary file matches
1653
```

9.

Ans:27423 times “Australia” appeared in the message columns.

Code: $ awk -F ',' '{print $6}' FIT1043_Dataset | grep "Australia" -i | wc -w

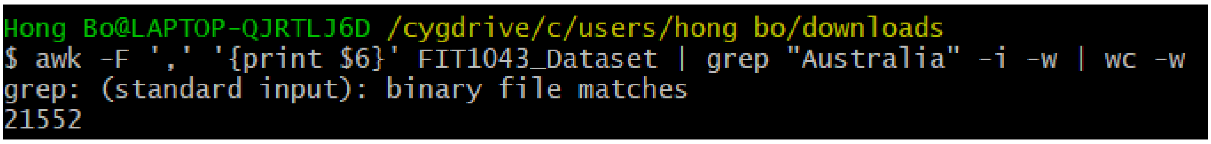


```
Hong Bo@LAPTOP-QJRTLJ6D /cygdrive/c/users/hong bo/downloads
$ awk -F ',' '{print $6}' FIT1043_Dataset | grep "Australia" -i | wc -w
grep: (standard input): binary file matches
27423
```

10.

Ans: 21552 words exactly the same as the word “Australia”.

Code: $ awk -F ',' '{print $6}' FIT1043_Dataset | grep "Australia" -w -i | wc -w

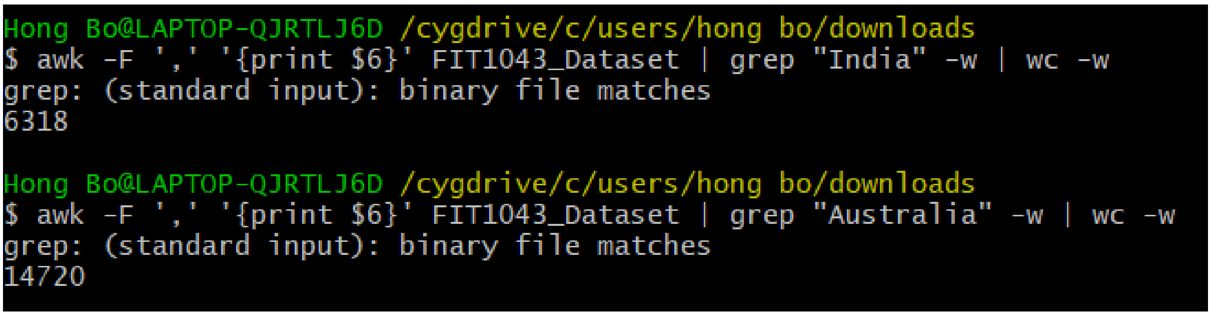


```
Hong Bo@LAPTOP-QJRTLJ6D /cygdrive/c/users/hong bo/downloads
$ awk -F ',' '{print $6}' FIT1043_Dataset | grep "Australia" -i -w | wc -w
grep: (standard input): binary file matches
21552
```

11.

Ans: As we can see Australia is more popular since it is frequently called in this tweet.

Code:$ awk -F ',' '{print $6}' FIT1043_Dataset | grep "India" -w | wc -w

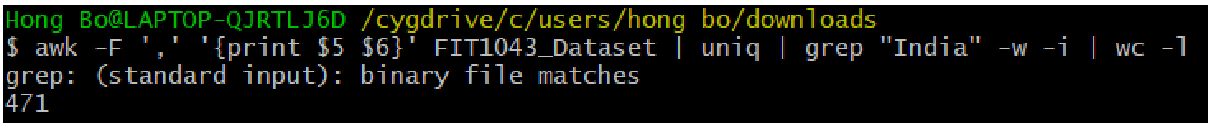


```
Hong Bo@LAPTOP-QJRTLJ6D /cygdrive/c/users/hong bo/downloads
$ awk -F ',' '{print $6}' FIT1043_Dataset | grep "India" -w | wc -w
grep: (standard input): binary file matches
6318

Hong Bo@LAPTOP-QJRTLJ6D /cygdrive/c/users/hong bo/downloads
$ awk -F ',' '{print $6}' FIT1043_Dataset | grep "Australia" -w | wc -w
grep: (standard input): binary file matches
14720
```

12.

Ans:471 unique users who mentioned the word India.

Code:$ awk -F ',' '{print $5 $6}' FIT1043_Dataset | uniq | grep "India" -w -i | wc -l



```
Hong Bo@LAPTOP-QJRTLJ6D /cygdrive/c/users/hong bo/downloads
$ awk -F ',' '{print $5 $6}' FIT1043_Dataset | uniq | grep "India" -w -i | wc -l
grep: (standard input): binary file matches
471
```

13.

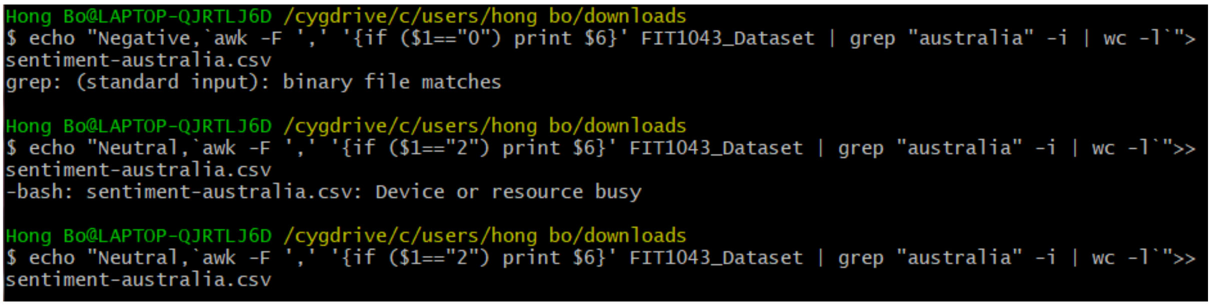
Ans: For this question, i will check if the first column of the file gets 0 which represent Negative,2 for Neutral,4 for Positive then by using echo we can make a sentence that has the word negative, positive then using backtick we can make a function call value saved in it then we can echo the value into the CSV files.

Code:

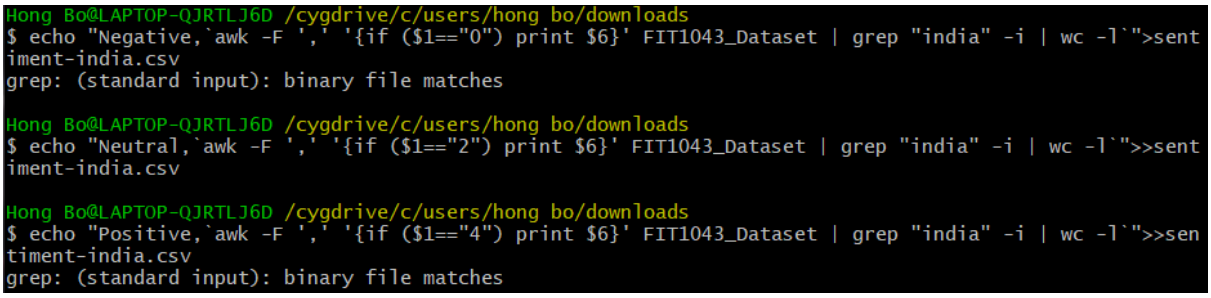
```
$ echo "Negative,`awk -F ',' '{if ($1=="0") print $6}' FIT1043_Dataset | grep "australia" -i | wc -l`">sentiment-australia.csv
$ echo "Neutral,`awk -F ',' '{if ($1=="2") print $6}' FIT1043_Dataset | grep "australia" -i | wc -l`">>sentiment-australia.csv
$ echo "Positive,`awk -F ',' '{if ($1=="4") print $6}' FIT1043_Dataset | grep "australia" -i | wc -l`">>sentiment-australia.csv
$ echo "Negative,`awk -F ',' '{if ($1=="0") print $6}' FIT1043_Dataset | grep "india" -i | wc -l`">sentiment-india.csv
$ echo "Neutral,`awk -F ',' '{if ($1=="2") print $6}' FIT1043_Dataset | grep "india" -i | wc -l`">>sentiment-india.csv
$ echo "Positive,`awk -F ',' '{if ($1=="4") print $6}' FIT1043_Dataset | grep "india" -i | wc -l`">>sentiment-india.csv
```

Checking:

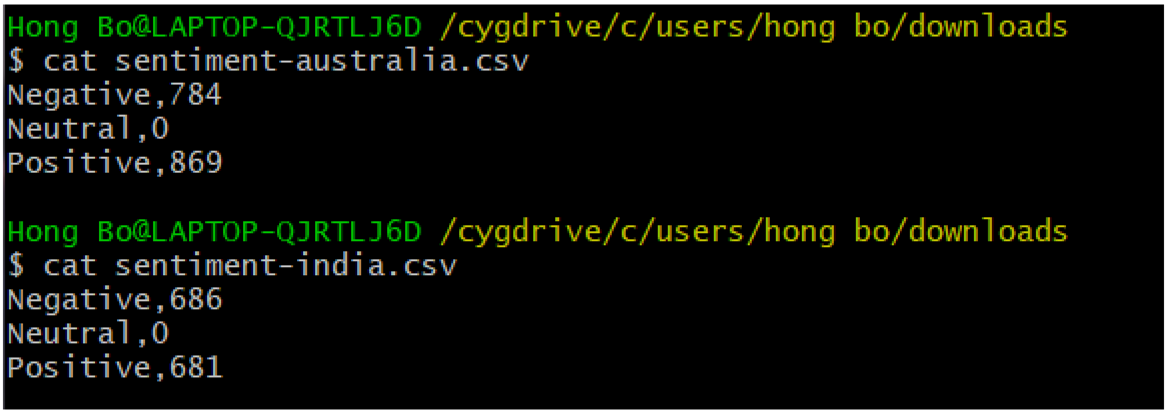
```
$ cat sentiment-australia.csv
$ cat sentiment-india.csv
```



```
Hong Bo@LAPTOP-QJRTLJ6D /cygdrive/c/users/hong bo/downloads
$ echo "Negative,`awk -F ',' '{if ($1=="0") print $6}' FIT1043_Dataset | grep "australia" -i | wc -l`">
sentiment-australia.csv
grep: (standard input): binary file matches

Hong Bo@LAPTOP-QJRTLJ6D /cygdrive/c/users/hong bo/downloads
$ echo "Neutral,`awk -F ',' '{if ($1=="2") print $6}' FIT1043_Dataset | grep "australia" -i | wc -l`">>
sentiment-australia.csv
-bash: sentiment-australia.csv: Device or resource busy

Hong Bo@LAPTOP-QJRTLJ6D /cygdrive/c/users/hong bo/downloads
$ echo "Neutral,`awk -F ',' '{if ($1=="2") print $6}' FIT1043_Dataset | grep "australia" -i | wc -l`">>
sentiment-australia.csv
```



```
Hong Bo@LAPTOP-QJRTLJ6D /cygdrive/c/users/hong bo/downloads
$ echo "Negative,`awk -F ',' '{if ($1=="0") print $6}' FIT1043_Dataset | grep "india" -i | wc -l`">sent
iment-india.csv
grep: (standard input): binary file matches

Hong Bo@LAPTOP-QJRTLJ6D /cygdrive/c/users/hong bo/downloads
$ echo "Neutral,`awk -F ',' '{if ($1=="2") print $6}' FIT1043_Dataset | grep "india" -i | wc -l`">>sent
iment-india.csv

Hong Bo@LAPTOP-QJRTLJ6D /cygdrive/c/users/hong bo/downloads
$ echo "Positive,`awk -F ',' '{if ($1=="4") print $6}' FIT1043_Dataset | grep "india" -i | wc -l`">>sen
timent-india.csv
grep: (standard input): binary file matches
```



```
Hong Bo@LAPTOP-QJRTLJ6D /cygdrive/c/users/hong bo/downloads
$ cat sentiment-australia.csv
Negative,784
Neutral,0
Positive,869

Hong Bo@LAPTOP-QJRTLJ6D /cygdrive/c/users/hong bo/downloads
$ cat sentiment-india.csv
Negative,686
Neutral,0
Positive,681
```

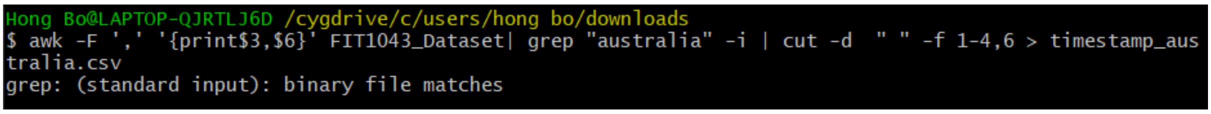
Part B

1.

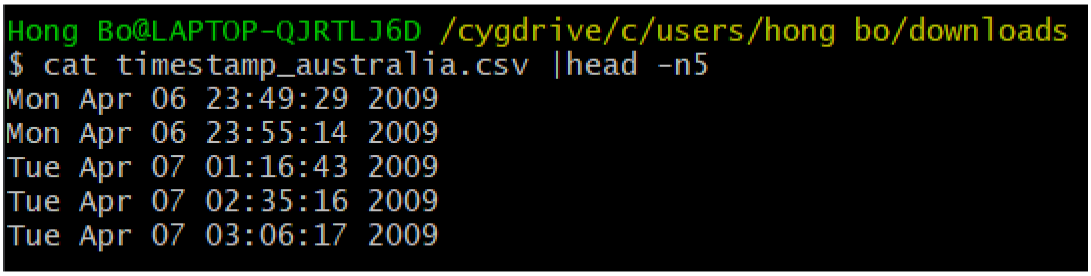
i)making a file of the timestamp_australia.csv by awk and cut and we will be butting of PDT cause in the next question there is no format for PDT so we will cut in 1-4,6.

Code:

$ awk -F ',' '{print$3,$6}' FIT1043_Dataset| grep "australia" -i | cut -d " " -f 1-4,6 > timestamp_australia.csv



```
Hong Bo@LAPTOP-QJRTLJ6D /cygdrive/c/users/hong bo/downloads
$ awk -F ',' '{print$3,$6}' FIT1043_Dataset| grep "australia" -i | cut -d  " " -f 1-4,6 > timestamp_aus
tralia.csv
grep: (standard input): binary file matches
```

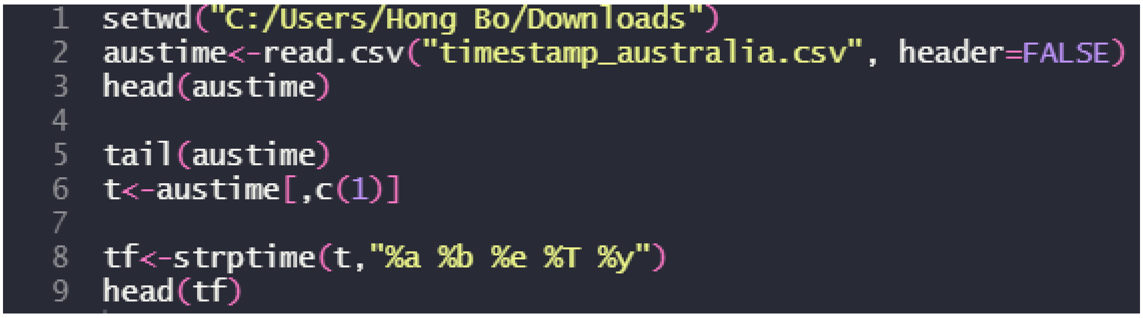


```
Hong Bo@LAPTOP-QJRTLJ6D /cygdrive/c/users/hong bo/downloads
$ cat timestamp_australia.csv |head -n5
Mon Apr 06 23:49:29 2009
Mon Apr 06 23:55:14 2009
Tue Apr 07 01:16:43 2009
Tue Apr 07 02:35:16 2009
Tue Apr 07 03:06:17 2009
```

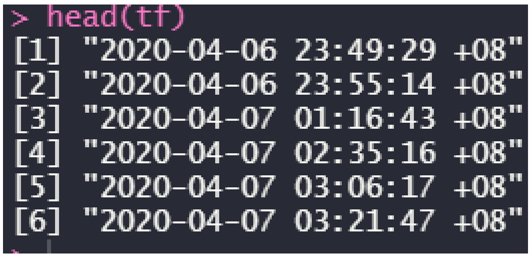
ii) As we can see, in the CSV file that was created in Cygwin there are the format of day month date hours minutes seconds, and years. These all dates and times can be told in the format which is in the order %a %b %e %T %y

Code:

```
austime<-read.csv("timestamp_australia.csv", header=FALSE)
head(austime)
tail(austime)
t<-austime[,c(1)]
tf<-strptime(t,"%a %b %e %T %y")
head(tf)
```



```
setwd("C:/Users/Hong Bo/Downloads")
austime<-read.csv("timestamp_australia.csv", header=FALSE)
head(austime)

tail(austime)
t<-austime[,c(1)]

tf<-strptime(t,"%a %b %e %T %y")
head(tf)
```

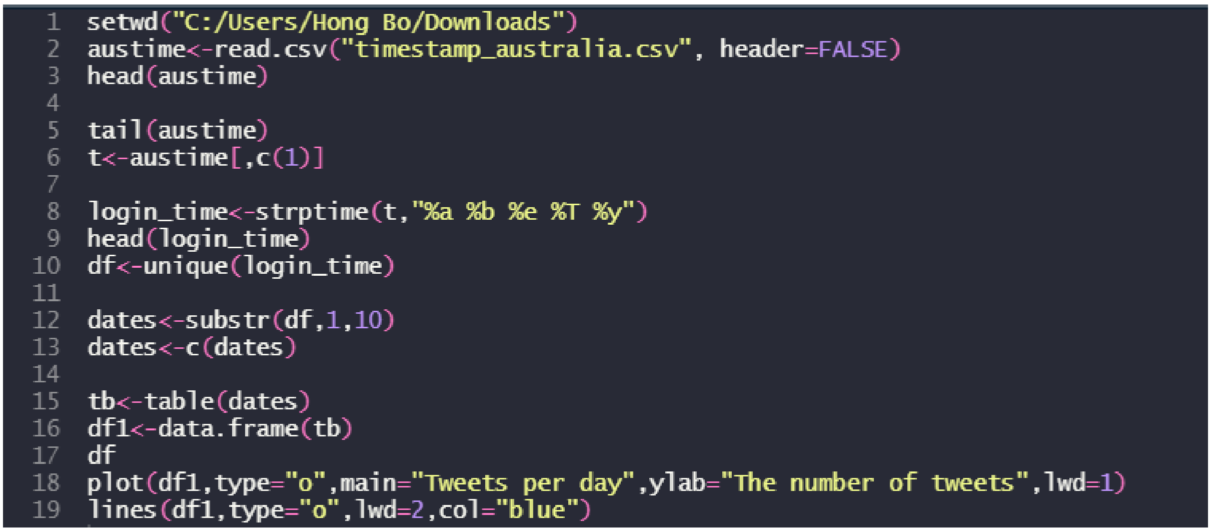


```
> head(tf)
[1] "2020-04-06 23:49:29 +08"
[2] "2020-04-06 23:55:14 +08"
[3] "2020-04-07 01:16:43 +08"
[4] "2020-04-07 02:35:16 +08"
[5] "2020-04-07 03:06:17 +08"
[6] "2020-04-07 03:21:47 +08"
```

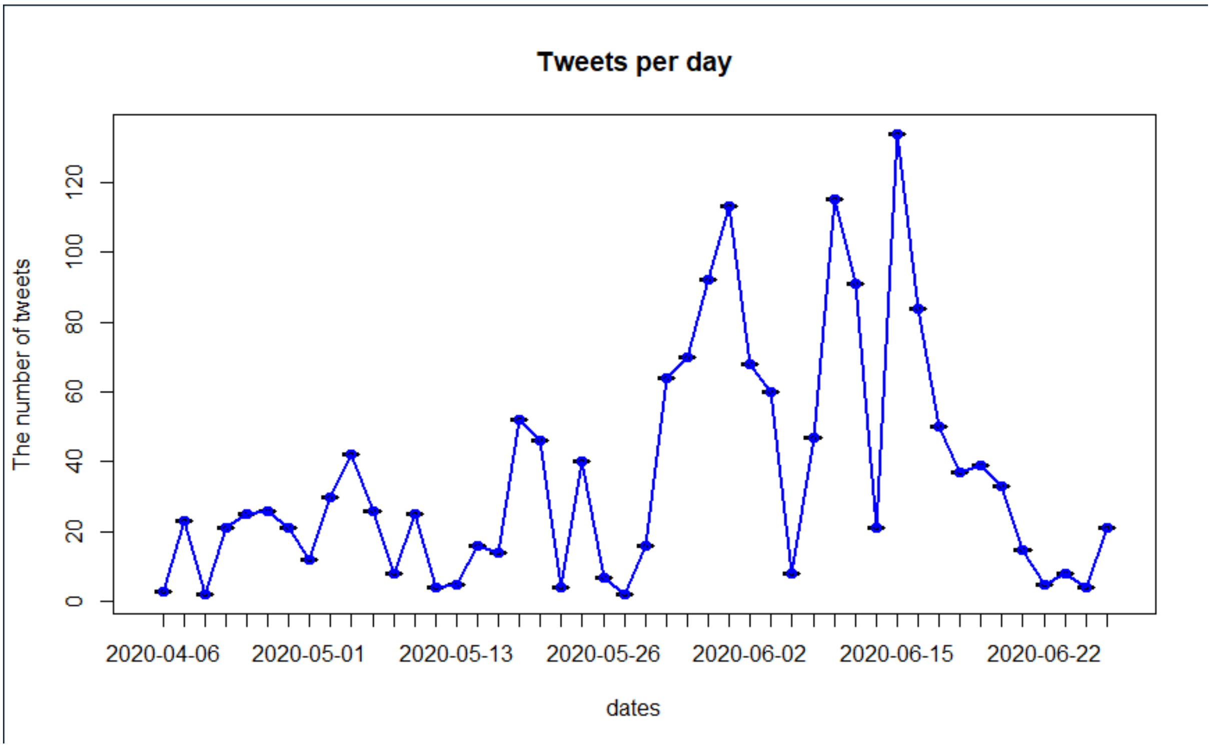
iii)

Code:

```
login_time<-strptime(t,"%a %b %e %T %y")
head(login_time)
df<-unique(login_time)

dates<-substr(df,1,10)
dates<-c(dates)

tb<-table(dates)
df1<-data.frame(tb)
df
plot(df1,type="o",main="Tweets per day",ylab="The number of tweets",lwd=1)
lines(df1,type="o",lwd=2,col="blue")
```



```
setwd("C:/Users/Hong Bo/Downloads")
austime<-read.csv("timestamp_australia.csv", header=FALSE)
head(austime)

tail(austime)
t<-austime[,c(1)]

login_time<-strptime(t,"%a %b %e %T %y")
head(login_time)
df<-unique(login_time)

dates<-substr(df,1,10)
dates<-c(dates)

tb<-table(dates)
df1<-data.frame(tb)
df
plot(df1,type="o",main="Tweets per day",ylab="The number of tweets",lwd=1)
lines(df1,type="o",lwd=2,col="blue")
```


Tweets per day
The number of tweets
0
20
40
60
80
100
120
dates
2020-04-06
2020-05-01
2020-05-13
2020-05-26
2020-06-02
2020-06-15
2020-06-22