Homework #2 (Homework Group 4)

Question 1:

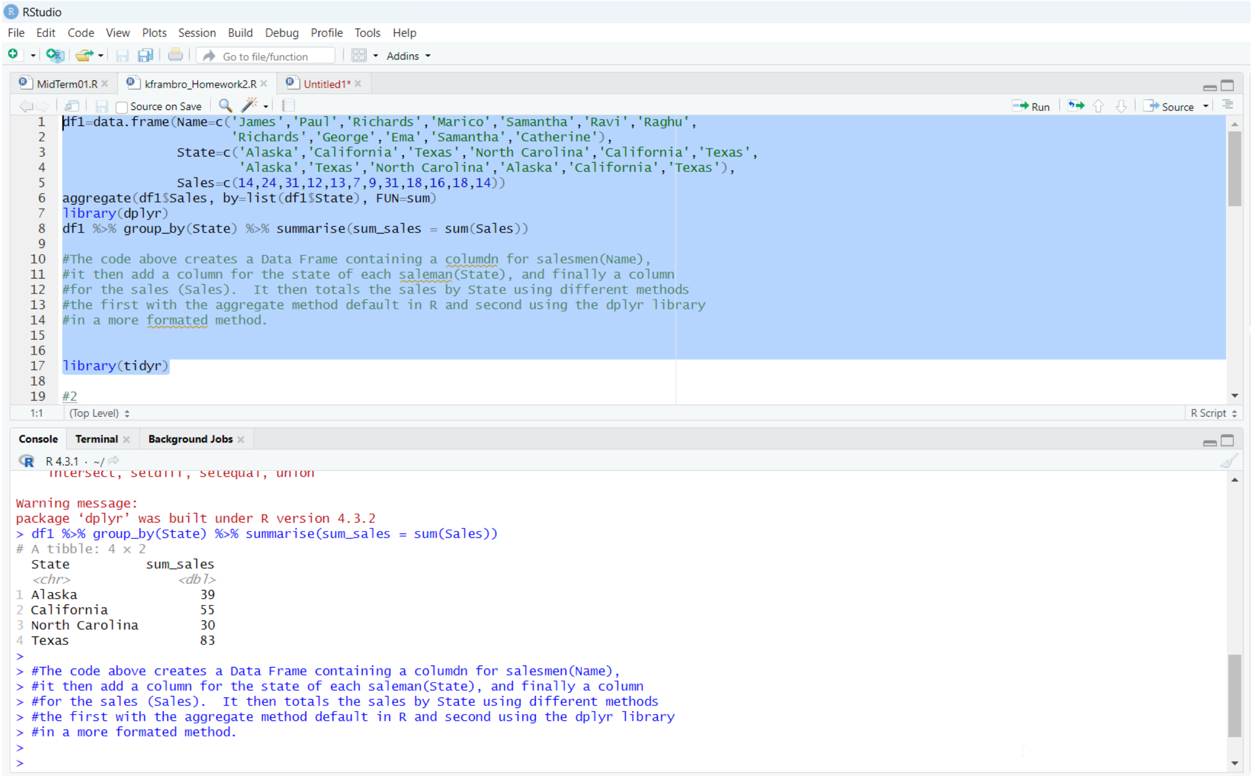
#The code above creates a Data Frame containing a columdn for salesmen(Name),

#it then add a column for the state of each saleman(State), and finally a column

#for the sales (Sales). It then totals the sales by State using different methods.

#the first with the aggregate method default in R and second using the dplyr library

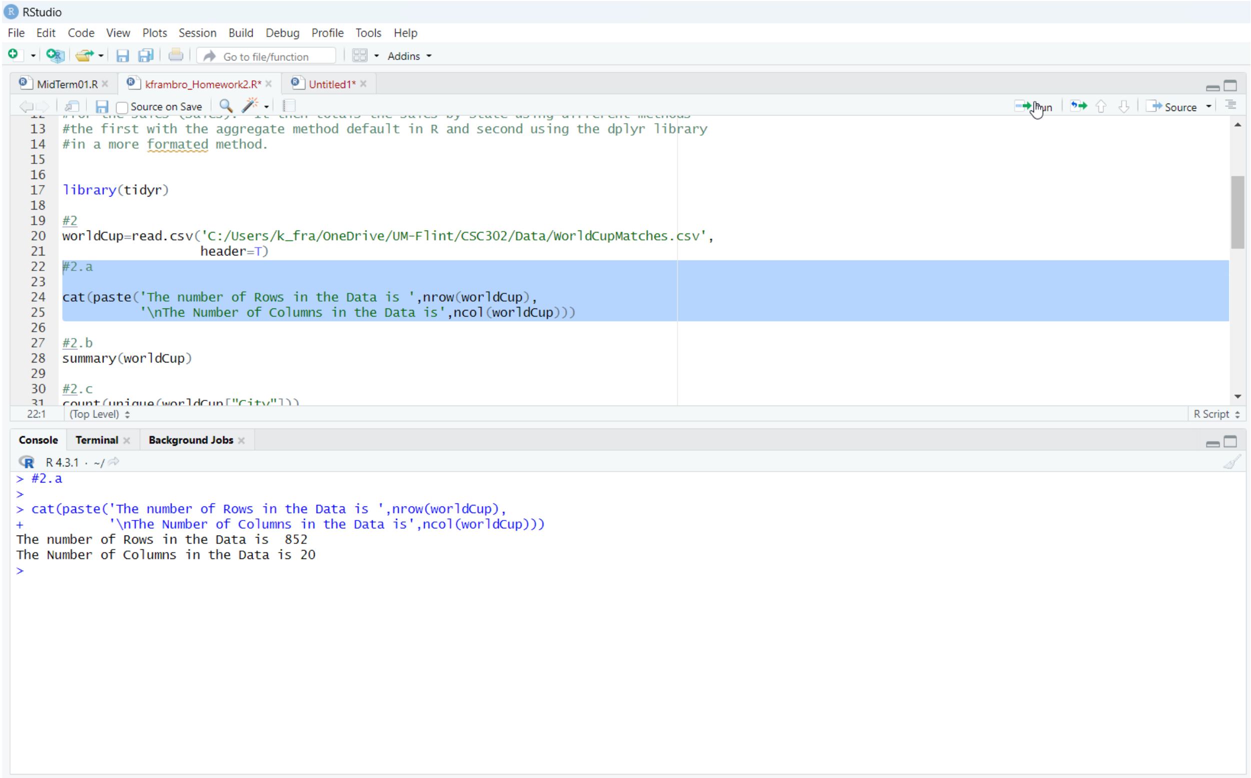
#in a more formated method.



```
df1=data.frame(Name=c('James','Paul','Richards','Marico','Samantha','Ravi','Raghu',
                      'Richards','George','Ema','Samantha','Catherine'),
               State=c('Alaska','California','Texas','North Carolina','California','Texas',
                       'Alaska','Texas','North Carolina','Alaska','California','Texas'),
               Sales=c(14,24,31,12,13,7,9,31,18,16,18,14))
aggregate(df1$Sales, by=list(df1$State), FUN=sum)
library(dplyr)
df1 %>% group_by(State) %>% summarise(sum_sales = sum(Sales))

#The code above creates a Data Frame containing a columdn for salesmen(Name),
#it then add a column for the state of each saleman(State), and finally a column
#for the sales (Sales).  It then totals the sales by State using different methods
#the first with the aggregate method default in R and second using the dplyr library
#in a more formated method.


library(tidyr)

#2
```

```
    intersect, setdiff, setequal, union

Warning message:
package ‘dplyr’ was built under R version 4.3.2
> df1 %>% group_by(State) %>% summarise(sum_sales = sum(Sales))
# A tibble: 4 × 2
  State          sum_sales
  <chr>              <dbl>
1 Alaska                39
2 California            55
3 North Carolina        30
4 Texas                 83
>
> #The code above creates a Data Frame containing a columdn for salesmen(Name),
> #it then add a column for the state of each saleman(State), and finally a column
> #for the sales (Sales).  It then totals the sales by State using different methods
> #the first with the aggregate method default in R and second using the dplyr library
> #in a more formated method.
>
>
```

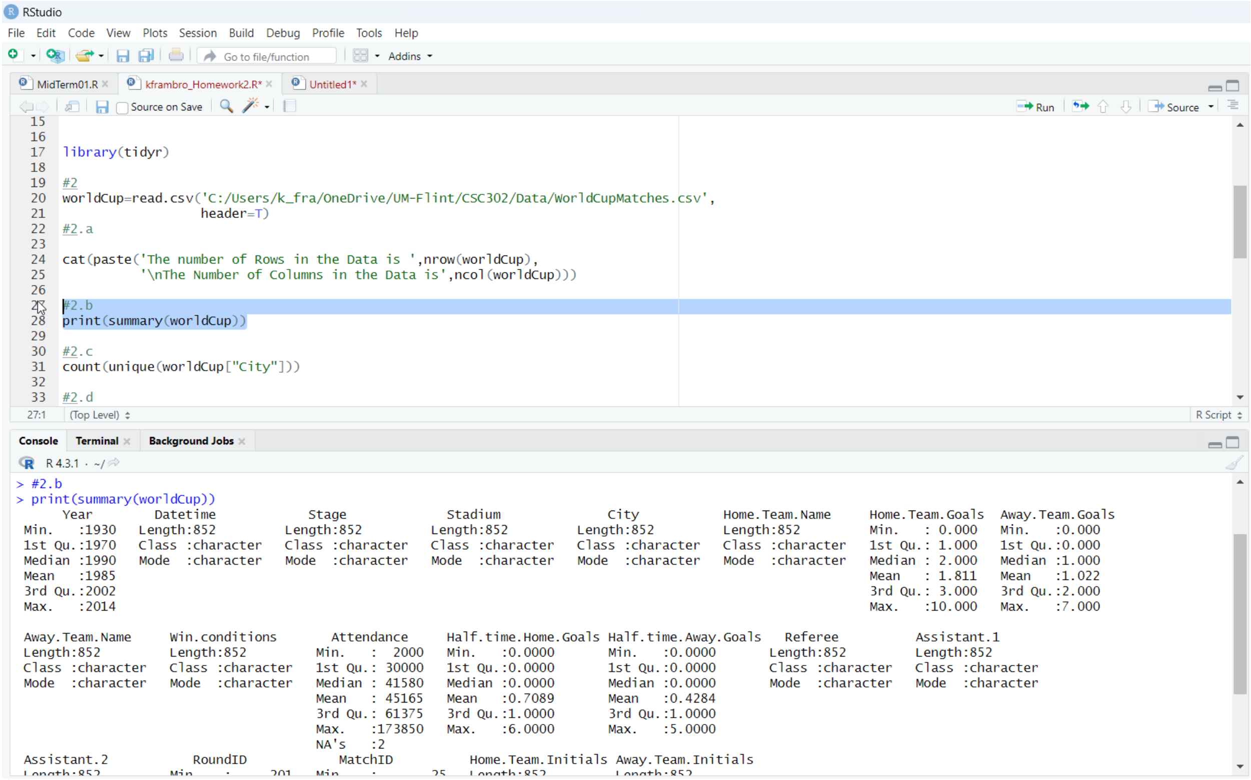
Question #2.A



```r
#for the sales (Sales). It then totals the sales by State using different methods
#the first with the aggregate method default in R and second using the dplyr library
#in a more formated method.


library(tidyr)

#2
worldCup=read.csv('C:/Users/k_fra/OneDrive/UM-Flint/CSC302/Data/WorldCupMatches.csv',
                  header=T)
#2.a

cat(paste('The number of Rows in the Data is ',nrow(worldCup),
          '\nThe Number of Columns in the Data is',ncol(worldCup)))

#2.b
summary(worldCup)

#2.c
count(unique(worldCup["City"]))
```

```
> #2.a
>
> cat(paste('The number of Rows in the Data is ',nrow(worldCup),
+           '\nThe Number of Columns in the Data is',ncol(worldCup)))
The number of Rows in the Data is  852
The Number of Columns in the Data is 20
>
```

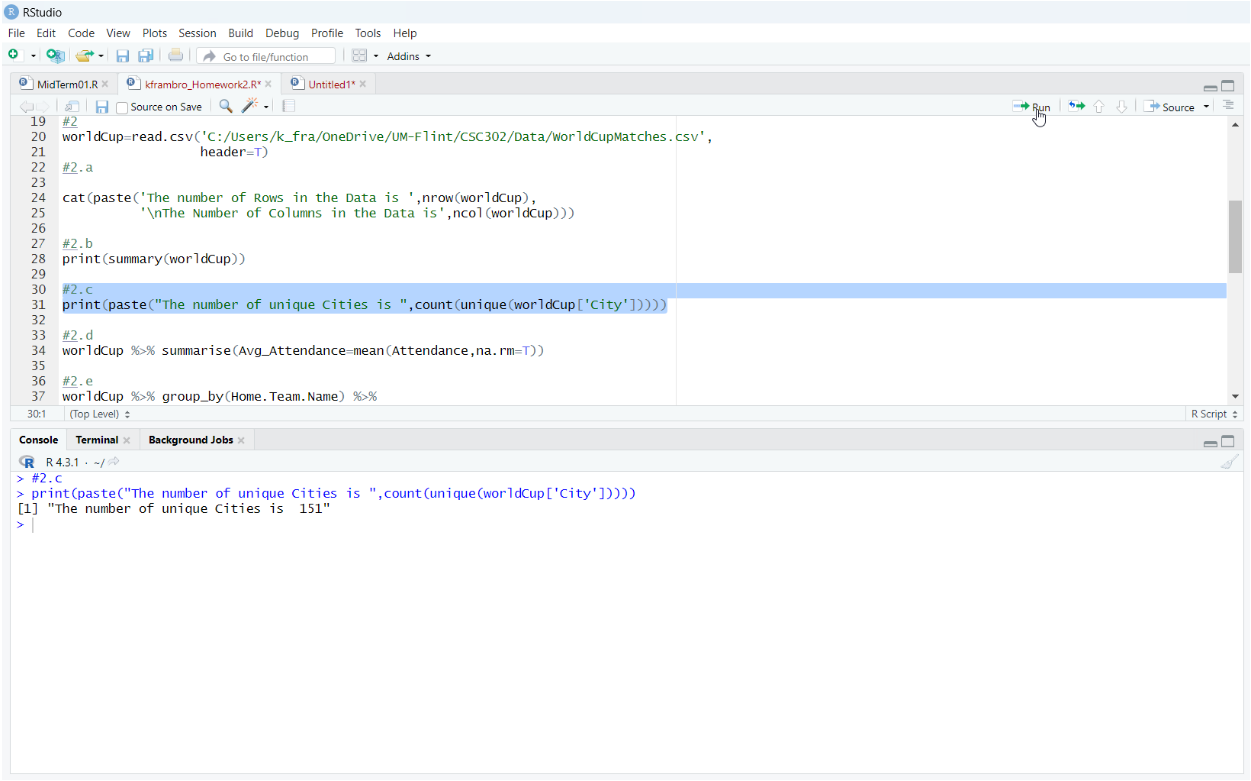
Question # 2.B



```r


library(tidyr)

#2
worldCup=read.csv('C:/Users/k_fra/OneDrive/UM-Flint/CSC302/Data/WorldCupMatches.csv',
                  header=T)
#2.a

cat(paste('The number of Rows in the Data is ',nrow(worldCup),
          '\nThe Number of Columns in the Data is',ncol(worldCup)))

#2.b
print(summary(worldCup))

#2.c
count(unique(worldCup["City"]))

#2.d
```

```
> #2.b
> print(summary(worldCup))
      Year        Datetime            Stage              Stadium              City           Home.Team.Name     Home.Team.Goals  Away.Team.Goals
 Min.   :1930   Length:852         Length:852         Length:852         Length:852         Length:852         Min.   : 0.000   Min.   :0.000  
 1st Qu.:1970   Class :character   Class :character   Class :character   Class :character   Class :character   1st Qu.: 1.000   1st Qu.:0.000  
 Median :1990   Mode  :character   Mode  :character   Mode  :character   Mode  :character   Mode  :character   Median : 2.000   Median :1.000  
 Mean   :1985                                                                                                  Mean   : 1.811   Mean   :1.022  
 3rd Qu.:2002                                                                                                  3rd Qu.: 3.000   3rd Qu.:2.000  
 Max.   :2014                                                                                                  Max.   :10.000   Max.   :7.000  

 Away.Team.Name     Win.conditions       Attendance     Half.time.Home.Goals Half.time.Away.Goals   Referee           Assistant.1       
 Length:852         Length:852         Min.   :  2000   Min.   :0.0000       Min.   :0.0000       Length:852         Length:852        
 Class :character   Class :character   1st Qu.: 30000   1st Qu.:0.0000       1st Qu.:0.0000       Class :character   Class :character  
 Mode  :character   Mode  :character   Median : 41580   Median :0.0000       Median :0.0000       Mode  :character   Mode  :character  
                                       Mean   : 45165   Mean   :0.7089       Mean   :0.4284                                            
                                       3rd Qu.: 61375   3rd Qu.:1.0000       3rd Qu.:1.0000                                            
                                       Max.   :173850   Max.   :6.0000       Max.   :5.0000                                            
                                       NA's   :2                                                                                       
 Assistant.2           RoundID            MatchID           Home.Team.Initials Away.Team.Initials
 Length:852         Min.   :     201   Min.   :      25   Length:852         Length:852        
```

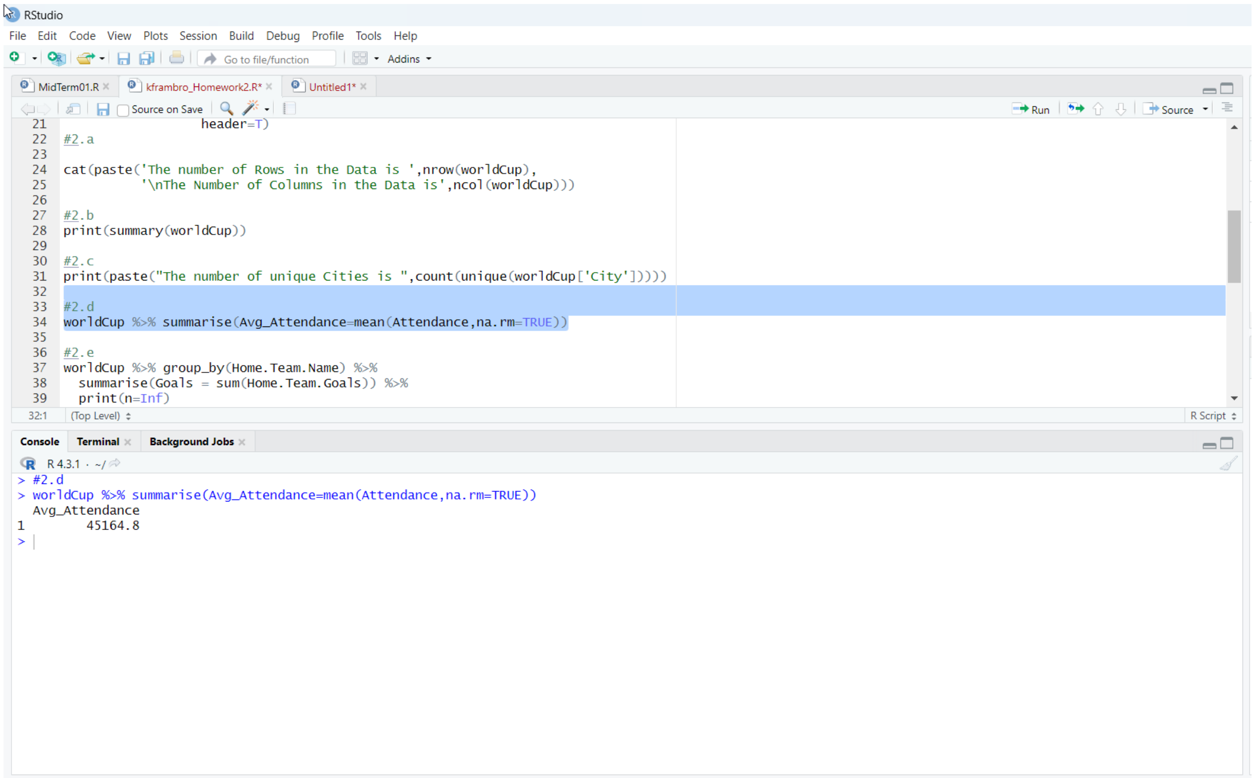
Question #2.C



```r
#2
worldCup=read.csv('C:/Users/k_fra/OneDrive/UM-Flint/CSC302/Data/WorldCupMatches.csv',
                  header=T)
#2.a

cat(paste('The number of Rows in the Data is ',nrow(worldCup),
          '\nThe Number of Columns in the Data is',ncol(worldCup)))

#2.b
print(summary(worldCup))

#2.c
print(paste("The number of unique Cities is ",count(unique(worldCup['City']))))

#2.d
worldCup %>% summarise(Avg_Attendance=mean(Attendance,na.rm=T))

#2.e
worldCup %>% group_by(Home.Team.Name) %>%
```

```
> #2.c
> print(paste("The number of unique Cities is ",count(unique(worldCup['City']))))
[1] "The number of unique Cities is  151"
> 
```

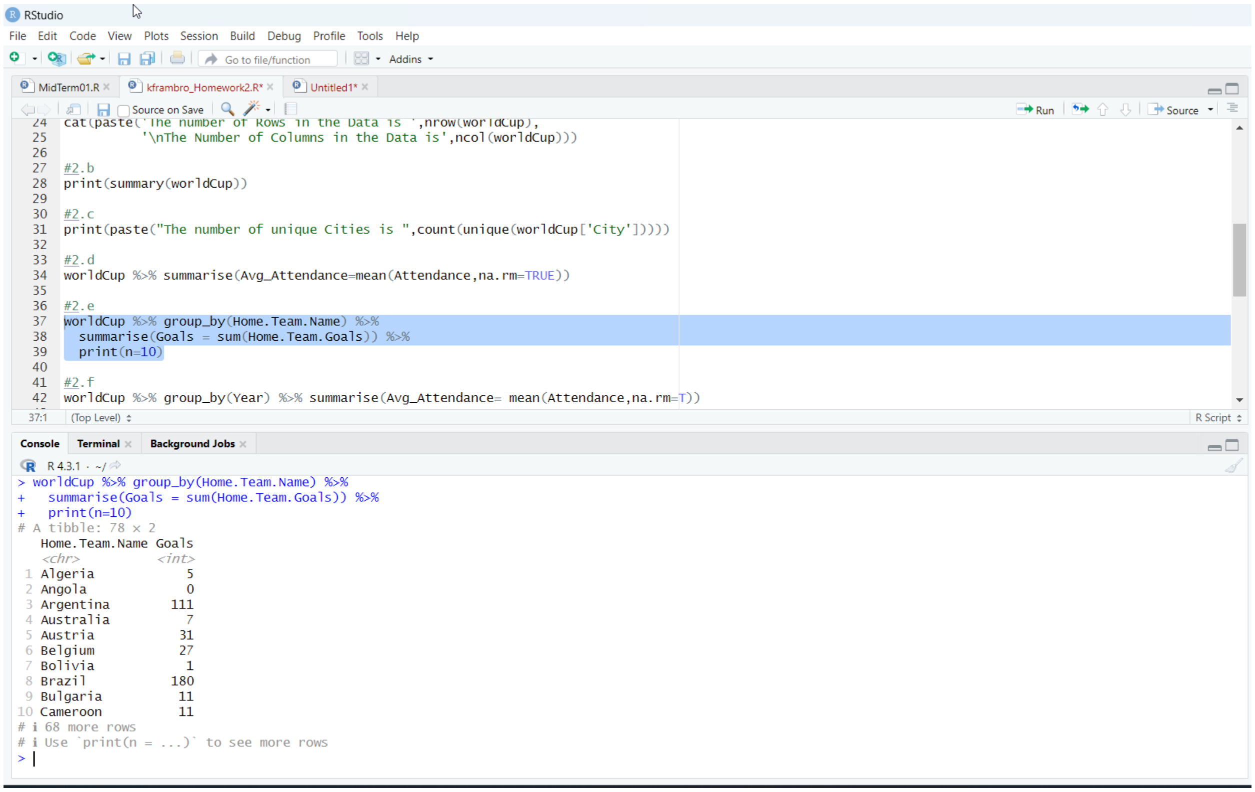
Question # 2.D



```
                  header=T)
#2.a

cat(paste('The number of Rows in the Data is ',nrow(worldCup),
          '\nThe Number of Columns in the Data is',ncol(worldCup)))

#2.b
print(summary(worldCup))

#2.c
print(paste("The number of unique Cities is ",count(unique(worldCup['City']))))

#2.d
worldCup %>% summarise(Avg_Attendance=mean(Attendance,na.rm=TRUE))

#2.e
worldCup %>% group_by(Home.Team.Name) %>%
  summarise(Goals = sum(Home.Team.Goals)) %>%
  print(n=Inf)
```

```
> #2.d
> worldCup %>% summarise(Avg_Attendance=mean(Attendance,na.rm=TRUE))
  Avg_Attendance
1        45164.8
> 
```

Question #2.E

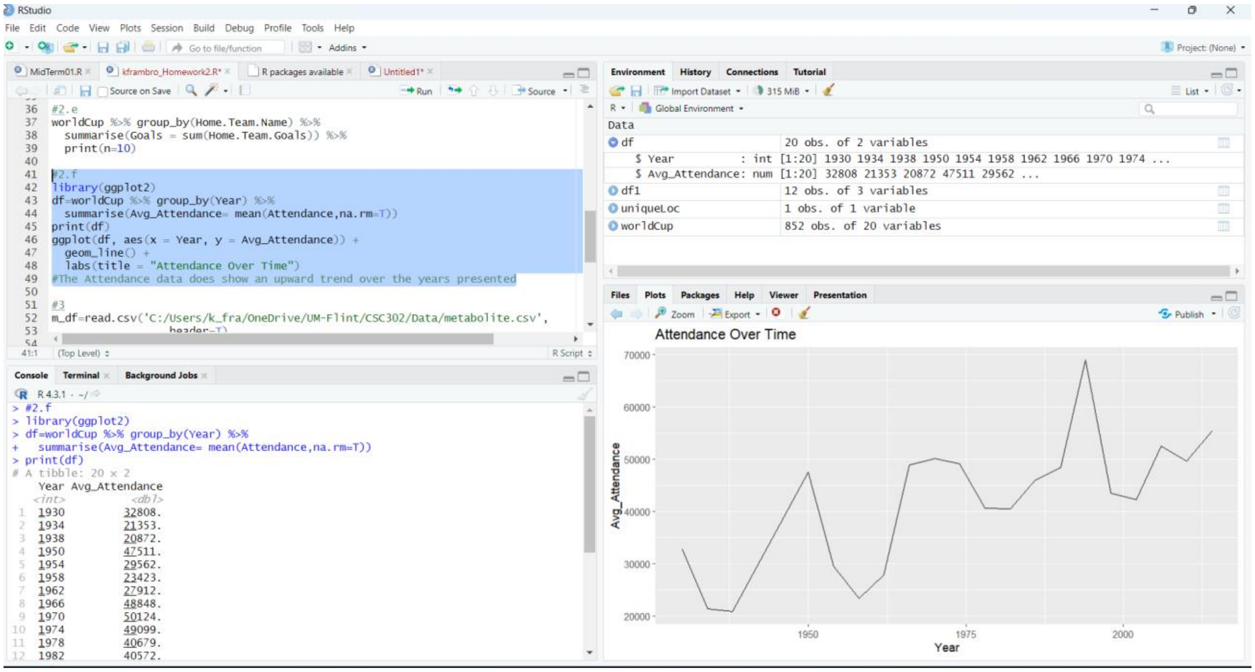


```
cat(paste('The number of Rows in the Data is ',nrow(worldCup),
          '\nThe Number of Columns in the Data is',ncol(worldCup)))

#2.b
print(summary(worldCup))

#2.c
print(paste("The number of unique Cities is ",count(unique(worldCup['City']))))

#2.d
worldCup %>% summarise(Avg_Attendance=mean(Attendance,na.rm=TRUE))

#2.e
worldCup %>% group_by(Home.Team.Name) %>%
  summarise(Goals = sum(Home.Team.Goals)) %>%
  print(n=10)

#2.f
worldCup %>% group_by(Year) %>% summarise(Avg_Attendance= mean(Attendance,na.rm=T))
```

```
> worldCup %>% group_by(Home.Team.Name) %>%
+   summarise(Goals = sum(Home.Team.Goals)) %>%
+   print(n=10)
# A tibble: 78 × 2
   Home.Team.Name Goals
   <chr>          <int>
 1 Algeria            5
 2 Angola             0
 3 Argentina        111
 4 Australia          7
 5 Austria           31
 6 Belgium           27
 7 Bolivia            1
 8 Brazil           180
 9 Bulgaria          11
10 Cameroon          11
# ℹ 68 more rows
# ℹ Use `print(n = ...)` to see more rows
> 
```

Question # 2.F

The Data does show an overall upward (increasing attendance) over the years presented

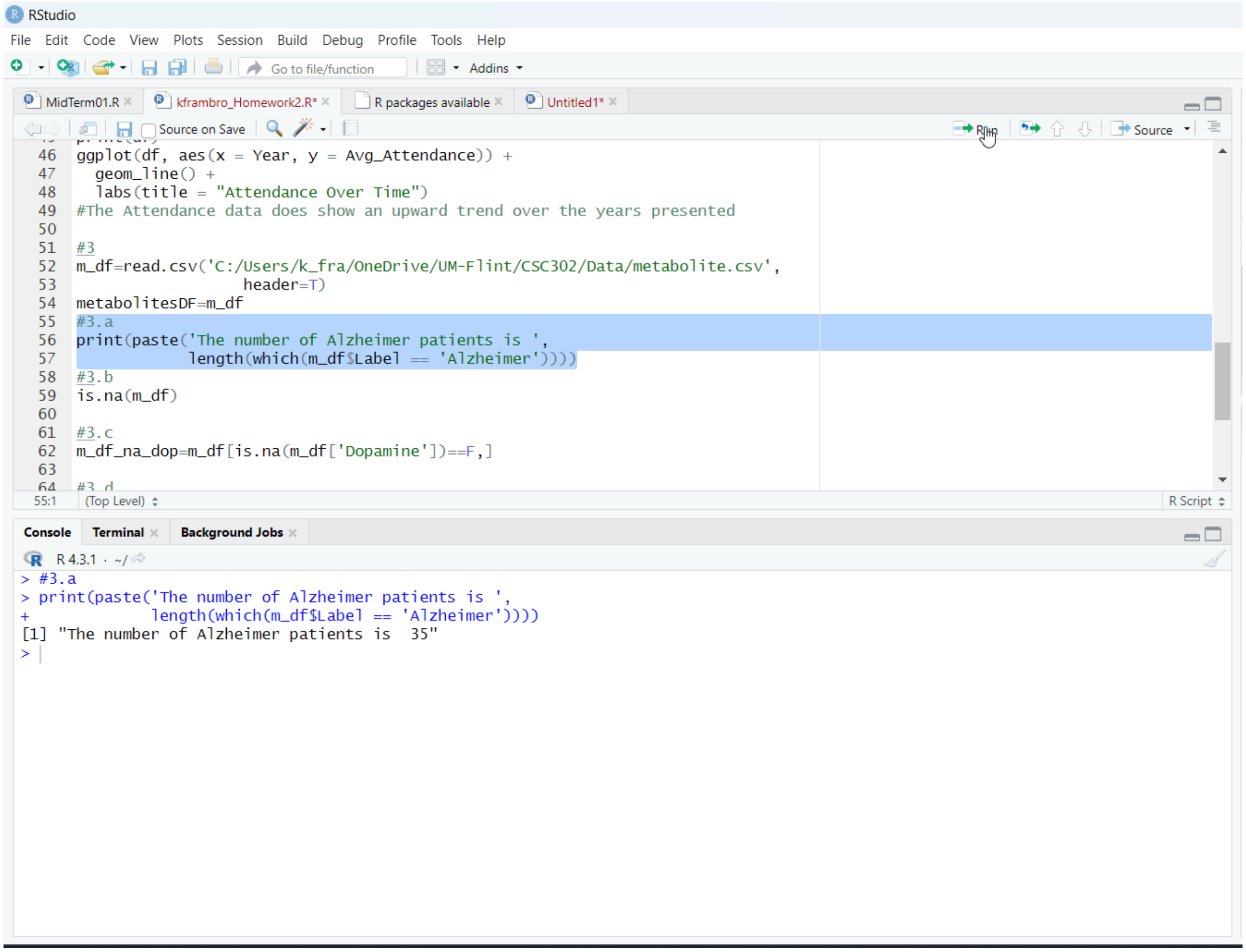


```r
#2.e
worldCup %>% group_by(Home.Team.Name) %>%
  summarise(Goals = sum(Home.Team.Goals)) %>%
  print(n=10)

#2.f
library(ggplot2)
df=worldCup %>% group_by(Year) %>%
  summarise(Avg_Attendance= mean(Attendance,na.rm=T))
print(df)
ggplot(df, aes(x = Year, y = Avg_Attendance)) +
  geom_line() +
  labs(title = "Attendance Over Time")
#The Attendance data does show an upward trend over the years presented

#3
m_df=read.csv('C:/Users/k_fra/OneDrive/UM-Flint/CSC302/Data/metabolite.csv',
                  header=T)
```

```
Data
df                   20 obs. of 2 variables
    $ Year          : int [1:20] 1930 1934 1938 1950 1954 1958 1962 1966 1970 1974 ...
    $ Avg_Attendance: num [1:20] 32808 21353 20872 47511 29562 ...
df1                  12 obs. of 3 variables
uniqueLoc            1 obs. of 1 variable
worldCup             852 obs. of 20 variables
```

```
> #2.f
> library(ggplot2)
> df=worldCup %>% group_by(Year) %>%
+   summarise(Avg_Attendance= mean(Attendance,na.rm=T))
> print(df)
# A tibble: 20 × 2
    Year Avg_Attendance
   <int>          <dbl>
 1  1930         32808.
 2  1934         21353.
 3  1938         20872.
 4  1950         47511.
 5  1954         29562.
 6  1958         23423.
 7  1962         27912.
 8  1966         48848.
 9  1970         50124.
10  1974         49099.
11  1978         40679.
12  1982         40572.
```

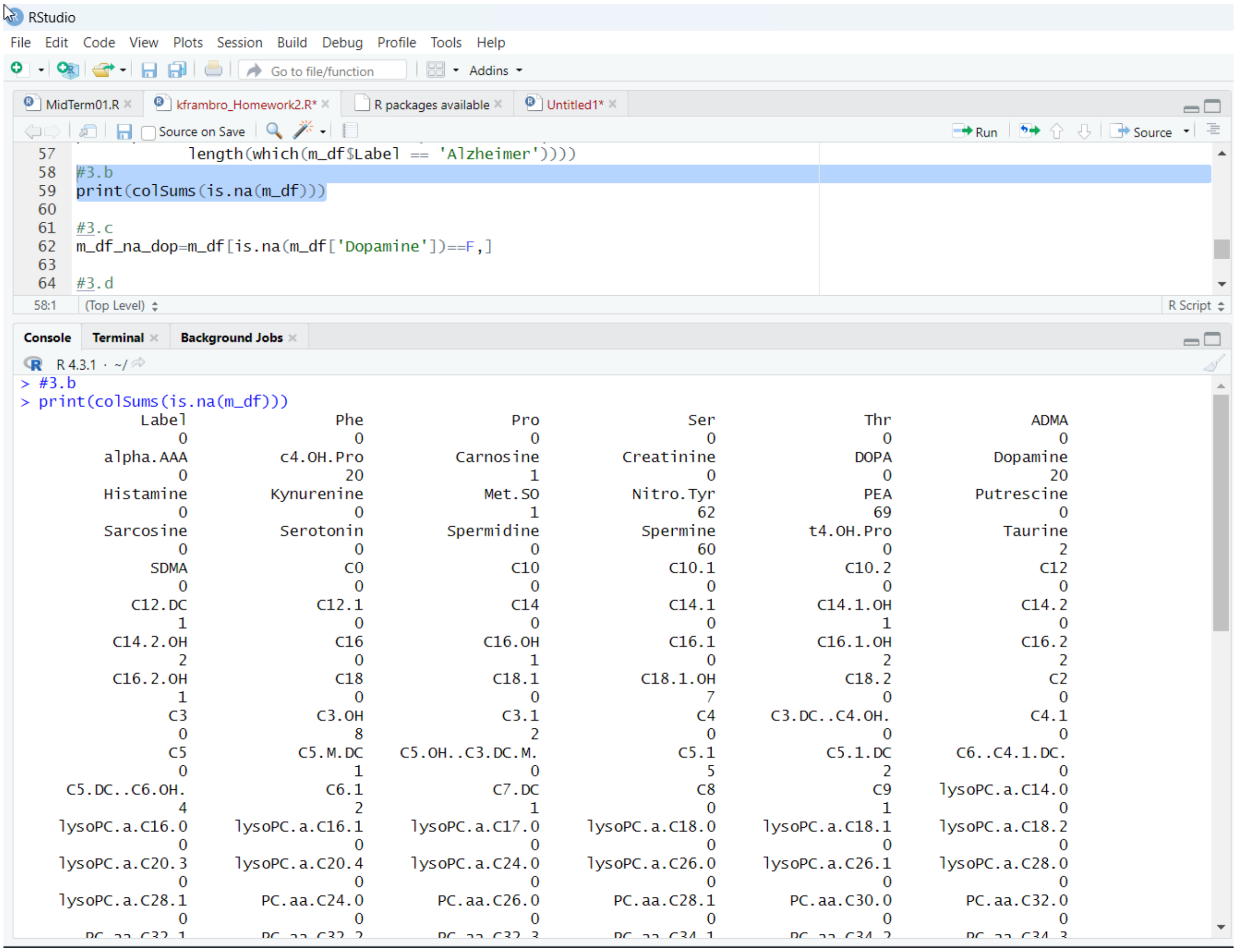
Question # 3.A



```r
ggplot(df, aes(x = Year, y = Avg_Attendance)) +
  geom_line() +
  labs(title = "Attendance Over Time")
#The Attendance data does show an upward trend over the years presented

#3
m_df=read.csv('C:/Users/k_fra/OneDrive/UM-Flint/CSC302/Data/metabolite.csv',
                  header=T)
metabolitesDF=m_df
#3.a
print(paste('The number of Alzheimer patients is ',
            length(which(m_df$Label == 'Alzheimer'))))
#3.b
is.na(m_df)

#3.c
m_df_na_dop=m_df[is.na(m_df['Dopamine'])==F,]

#3.d
```

```
> #3.a
> print(paste('The number of Alzheimer patients is ',
+             length(which(m_df$Label == 'Alzheimer'))))
[1] "The number of Alzheimer patients is  35"
> 
```

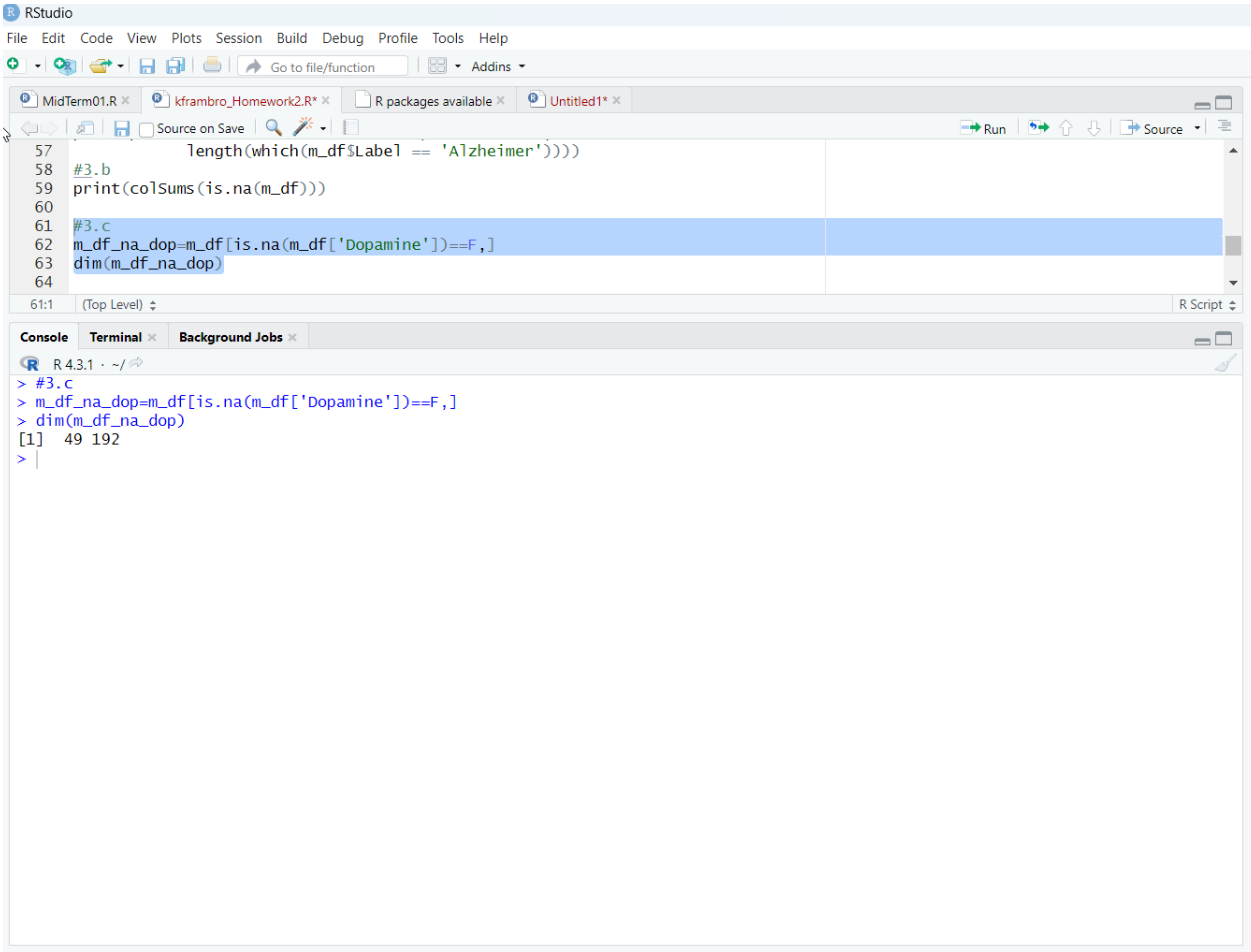
Question #3.B



```
                length(which(m_df$Label == 'Alzheimer'))))
#3.b
print(colSums(is.na(m_df)))

#3.c
m_df_na_dop=m_df[is.na(m_df['Dopamine'])==F,]

#3.d
```

```
> #3.b
> print(colSums(is.na(m_df)))
          Label             Phe             Pro             Ser             Thr            ADMA 
              0               0               0               0               0               0 
      alpha.AAA       c4.OH.Pro       Carnosine      Creatinine            DOPA        Dopamine 
              0              20               1               0               0              20 
      Histamine      Kynurenine          Met.SO       Nitro.Tyr             PEA      Putrescine 
              0               0               1              62              69               0 
      Sarcosine       Serotonin      Spermidine        Spermine       t4.OH.Pro         Taurine 
              0               0               0              60               0               2 
           SDMA              C0             C10           C10.1           C10.2             C12 
              0               0               0               0               0               0 
         C12.DC           C12.1             C14           C14.1        C14.1.OH           C14.2 
              1               0               0               0               1               0 
       C14.2.OH             C16          C16.OH           C16.1        C16.1.OH           C16.2 
              2               0               1               0               2               2 
       C16.2.OH             C18           C18.1        C18.1.OH           C18.2              C2 
              1               0               0               7               0               0 
             C3           C3.OH            C3.1              C4   C3.DC..C4.OH.            C4.1 
              0               8               2               0               0               0 
             C5         C5.M.DC C5.OH..C3.DC.M.            C5.1         C5.1.DC    C6..C4.1.DC. 
              0               1               0               5               2               0 
  C5.DC..C6.OH.            C6.1           C7.DC              C8              C9  lysoPC.a.C14.0 
              4               2               1               0               1               0 
 lysoPC.a.C16.0  lysoPC.a.C16.1  lysoPC.a.C17.0  lysoPC.a.C18.0  lysoPC.a.C18.1  lysoPC.a.C18.2 
              0               0               0               0               0               0 
 lysoPC.a.C20.3  lysoPC.a.C20.4  lysoPC.a.C24.0  lysoPC.a.C26.0  lysoPC.a.C26.1  lysoPC.a.C28.0 
              0               0               0               0               0               0 
 lysoPC.a.C28.1     PC.aa.C24.0     PC.aa.C26.0     PC.aa.C28.1     PC.aa.C30.0     PC.aa.C32.0 
              0               0               0               0               0               0 
    PC.aa.C32.1     PC.aa.C32.2     PC.aa.C32.3     PC.aa.C34.1     PC.aa.C34.2     PC.aa.C34.3 
```

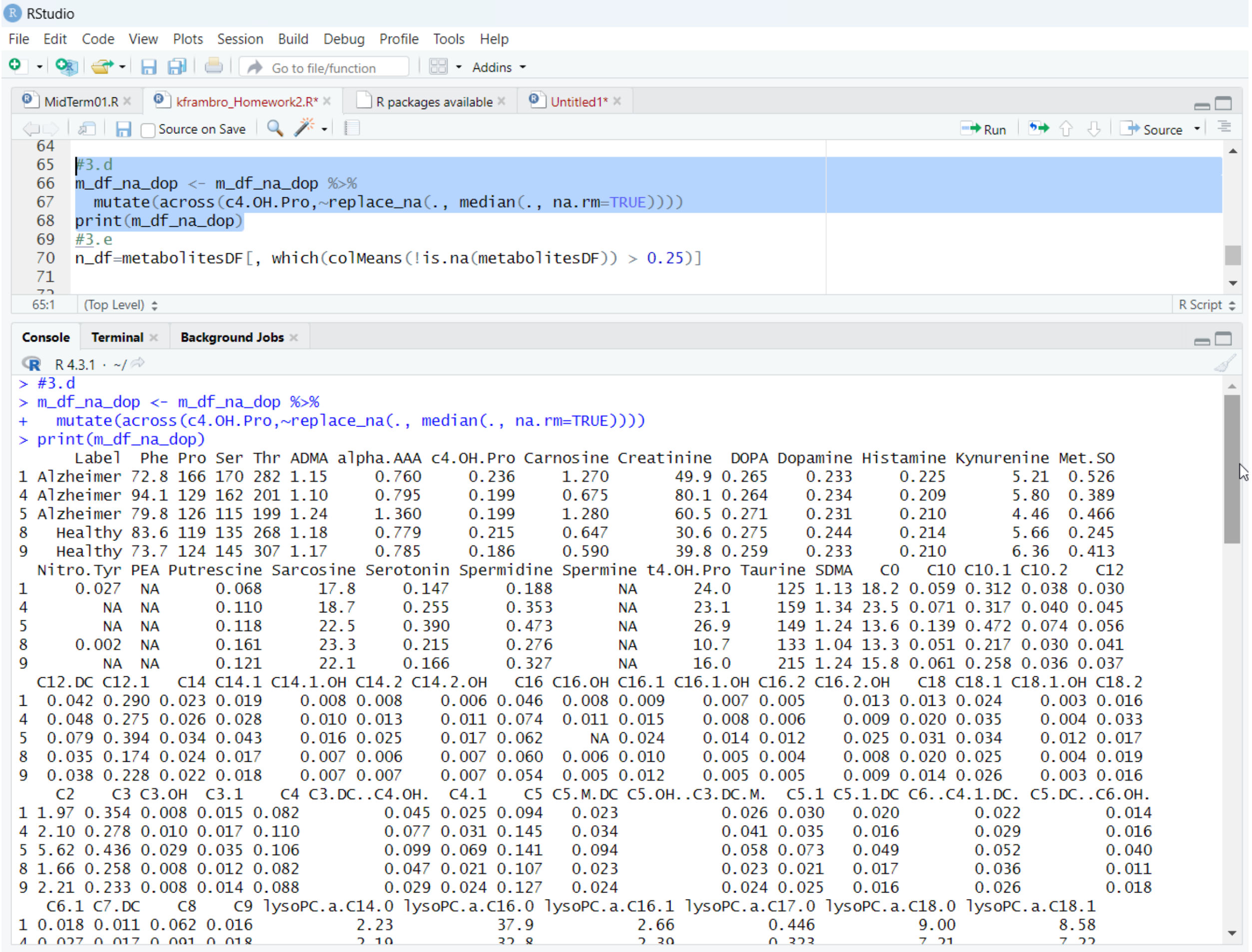
Question #3.C



```
            length(which(m_df$Label == 'Alzheimer'))))
#3.b
print(colSums(is.na(m_df)))

#3.c
m_df_na_dop=m_df[is.na(m_df['Dopamine'])==F,]
dim(m_df_na_dop)
```

```
> #3.c
> m_df_na_dop=m_df[is.na(m_df['Dopamine'])==F,]
> dim(m_df_na_dop)
[1]  49 192
> 
```

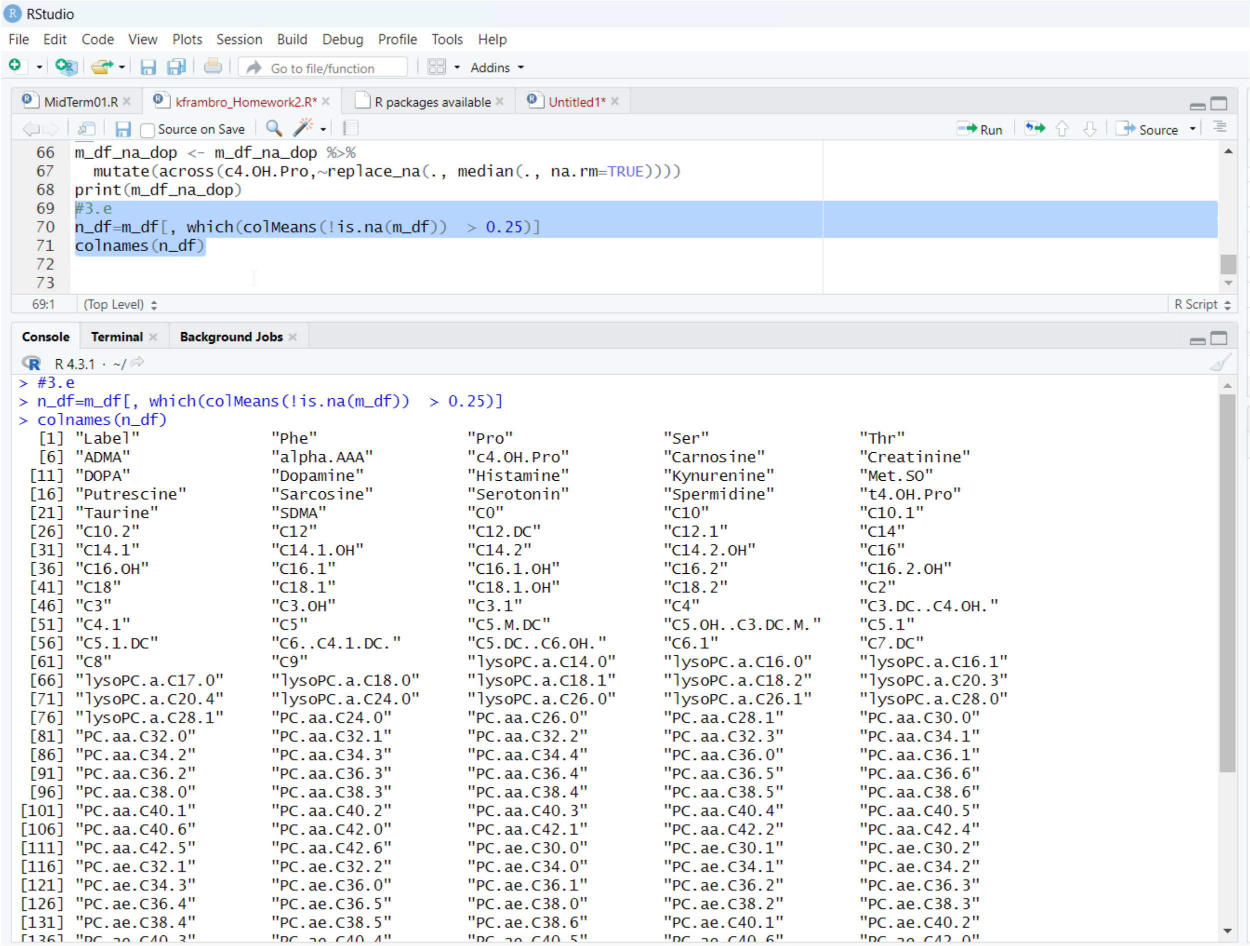
Question #3.D



```r
#3.d
m_df_na_dop <- m_df_na_dop %>%
  mutate(across(c4.OH.Pro,~replace_na(., median(., na.rm=TRUE))))
print(m_df_na_dop)
#3.e
n_df=metabolitesDF[, which(colMeans(!is.na(metabolitesDF)) > 0.25)]

```

```
> #3.d
> m_df_na_dop <- m_df_na_dop %>%
+   mutate(across(c4.OH.Pro,~replace_na(., median(., na.rm=TRUE))))
> print(m_df_na_dop)
      Label  Phe Pro Ser Thr ADMA alpha.AAA c4.OH.Pro Carnosine Creatinine  DOPA Dopamine Histamine Kynurenine Met.SO
1 Alzheimer 72.8 166 170 282 1.15     0.760     0.236     1.270       49.9 0.265    0.233     0.225       5.21  0.526
4 Alzheimer 94.1 129 162 201 1.10     0.795     0.199     0.675       80.1 0.264    0.234     0.209       5.80  0.389
5 Alzheimer 79.8 126 115 199 1.24     1.360     0.199     1.280       60.5 0.271    0.231     0.210       4.46  0.466
8   Healthy 83.6 119 135 268 1.18     0.779     0.215     0.647       30.6 0.275    0.244     0.214       5.66  0.245
9   Healthy 73.7 124 145 307 1.17     0.785     0.186     0.590       39.8 0.259    0.233     0.210       6.36  0.413
  Nitro.Tyr PEA Putrescine Sarcosine Serotonin Spermidine Spermine t4.OH.Pro Taurine SDMA   C0   C10 C10.1 C10.2   C12
1     0.027  NA      0.068      17.8     0.147      0.188       NA      24.0     125 1.13 18.2 0.059 0.312 0.038 0.030
4        NA  NA      0.110      18.7     0.255      0.353       NA      23.1     159 1.34 23.5 0.071 0.317 0.040 0.045
5        NA  NA      0.118      22.5     0.390      0.473       NA      26.9     149 1.24 13.6 0.139 0.472 0.074 0.056
8     0.002  NA      0.161      23.3     0.215      0.276       NA      10.7     133 1.04 13.3 0.051 0.217 0.030 0.041
9        NA  NA      0.121      22.1     0.166      0.327       NA      16.0     215 1.24 15.8 0.061 0.258 0.036 0.037
  C12.DC C12.1   C14 C14.1 C14.1.OH C14.2 C14.2.OH   C16 C16.OH C16.1 C16.1.OH C16.2 C16.2.OH   C18 C18.1 C18.1.OH C18.2
1  0.042 0.290 0.023 0.019    0.008 0.008    0.006 0.046  0.008 0.009    0.007 0.005    0.013 0.013 0.024    0.003 0.016
4  0.048 0.275 0.026 0.028    0.010 0.013    0.011 0.074  0.011 0.015    0.008 0.006    0.009 0.020 0.035    0.004 0.033
5  0.079 0.394 0.034 0.043    0.016 0.025    0.017 0.062     NA 0.024    0.014 0.012    0.025 0.031 0.034    0.012 0.017
8  0.035 0.174 0.024 0.017    0.007 0.006    0.007 0.060  0.006 0.010    0.005 0.004    0.008 0.020 0.025    0.004 0.019
9  0.038 0.228 0.022 0.018    0.007 0.007    0.007 0.054  0.005 0.012    0.005 0.005    0.009 0.014 0.026    0.003 0.016
    C2    C3 C3.OH  C3.1    C4 C3.DC..C4.OH.  C4.1    C5 C5.M.DC C5.OH..C3.DC.M.  C5.1 C5.1.DC C6..C4.1.DC. C5.DC..C6.OH.
1 1.97 0.354 0.008 0.015 0.082         0.045 0.025 0.094   0.023           0.026 0.030   0.020        0.022         0.014
4 2.10 0.278 0.010 0.017 0.110         0.077 0.031 0.145   0.034           0.041 0.035   0.016        0.029         0.016
5 5.62 0.436 0.029 0.035 0.106         0.099 0.069 0.141   0.094           0.058 0.073   0.049        0.052         0.040
8 1.66 0.258 0.008 0.012 0.082         0.047 0.021 0.107   0.023           0.023 0.021   0.017        0.036         0.011
9 2.21 0.233 0.008 0.014 0.088         0.029 0.024 0.127   0.024           0.024 0.025   0.016        0.026         0.018
   C6.1 C7.DC    C8    C9 lysoPC.a.C14.0 lysoPC.a.C16.0 lysoPC.a.C16.1 lysoPC.a.C17.0 lysoPC.a.C18.0 lysoPC.a.C18.1
1 0.018 0.011 0.062 0.016           2.23           37.9           2.66          0.446           9.00           8.58
4 0.027 0.017 0.091 0.018           2.19           32.8           2.39          0.323           7.21           7.22
```

Question 3.E



```r
m_df_na_dop <- m_df_na_dop %>%
  mutate(across(c4.OH.Pro,~replace_na(., median(., na.rm=TRUE))))
print(m_df_na_dop)
#3.e
n_df=m_df[, which(colMeans(!is.na(m_df))  > 0.25)]
colnames(n_df)
```

```
> #3.e
> n_df=m_df[, which(colMeans(!is.na(m_df))  > 0.25)]
> colnames(n_df)
  [1] "Label"              "Phe"                "Pro"                "Ser"                "Thr"
  [6] "ADMA"               "alpha.AAA"          "c4.OH.Pro"          "Carnosine"          "Creatinine"
 [11] "DOPA"               "Dopamine"           "Histamine"          "Kynurenine"         "Met.SO"
 [16] "Putrescine"         "Sarcosine"          "Serotonin"          "Spermidine"         "t4.OH.Pro"
 [21] "Taurine"            "SDMA"               "C0"                 "C10"                "C10.1"
 [26] "C10.2"              "C12"                "C12.DC"             "C12.1"              "C14"
 [31] "C14.1"              "C14.1.OH"           "C14.2"              "C14.2.OH"           "C16"
 [36] "C16.OH"             "C16.1"              "C16.1.OH"           "C16.2"              "C16.2.OH"
 [41] "C18"                "C18.1"              "C18.1.OH"           "C18.2"              "C2"
 [46] "C3"                 "C3.OH"              "C3.1"               "C4"                 "C3.DC..C4.OH."
 [51] "C4.1"               "C5"                 "C5.M.DC"            "C5.OH..C3.DC.M."    "C5.1"
 [56] "C5.1.DC"            "C6..C4.1.DC."       "C5.DC..C6.OH."      "C6.1"               "C7.DC"
 [61] "C8"                 "C9"                 "lysoPC.a.C14.0"     "lysoPC.a.C16.0"     "lysoPC.a.C16.1"
 [66] "lysoPC.a.C17.0"     "lysoPC.a.C18.0"     "lysoPC.a.C18.1"     "lysoPC.a.C18.2"     "lysoPC.a.C20.3"
 [71] "lysoPC.a.C20.4"     "lysoPC.a.C24.0"     "lysoPC.a.C26.0"     "lysoPC.a.C26.1"     "lysoPC.a.C28.0"
 [76] "lysoPC.a.C28.1"     "PC.aa.C24.0"        "PC.aa.C26.0"        "PC.aa.C28.1"        "PC.aa.C30.0"
 [81] "PC.aa.C32.0"        "PC.aa.C32.1"        "PC.aa.C32.2"        "PC.aa.C32.3"        "PC.aa.C34.1"
 [86] "PC.aa.C34.2"        "PC.aa.C34.3"        "PC.aa.C34.4"        "PC.aa.C36.0"        "PC.aa.C36.1"
 [91] "PC.aa.C36.2"        "PC.aa.C36.3"        "PC.aa.C36.4"        "PC.aa.C36.5"        "PC.aa.C36.6"
 [96] "PC.aa.C38.0"        "PC.aa.C38.3"        "PC.aa.C38.4"        "PC.aa.C38.5"        "PC.aa.C38.6"
[101] "PC.aa.C40.1"        "PC.aa.C40.2"        "PC.aa.C40.3"        "PC.aa.C40.4"        "PC.aa.C40.5"
[106] "PC.aa.C40.6"        "PC.aa.C42.0"        "PC.aa.C42.1"        "PC.aa.C42.2"        "PC.aa.C42.4"
[111] "PC.aa.C42.5"        "PC.aa.C42.6"        "PC.ae.C30.0"        "PC.ae.C30.1"        "PC.ae.C30.2"
[116] "PC.ae.C32.1"        "PC.ae.C32.2"        "PC.ae.C34.0"        "PC.ae.C34.1"        "PC.ae.C34.2"
[121] "PC.ae.C34.3"        "PC.ae.C36.0"        "PC.ae.C36.1"        "PC.ae.C36.2"        "PC.ae.C36.3"
[126] "PC.ae.C36.4"        "PC.ae.C36.5"        "PC.ae.C38.0"        "PC.ae.C38.2"        "PC.ae.C38.3"
[131] "PC.ae.C38.4"        "PC.ae.C38.5"        "PC.ae.C38.6"        "PC.ae.C40.1"        "PC.ae.C40.2"
[136] "PC.ae.C40.3"        "PC.ae.C40.4"        "PC.ae.C40.5"        "PC.ae.C40.6"        "PC.ae.C42.0"
```