```
R version 4.1.1 (2021-08-10) -- "Kick Things"
Copyright (C) 2021 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

  Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> # JRodoni_HW04_script.R
> # C:/Users/jackr/OneDrive/Desktop/Graduate School Courses/
> #   STAT 604 - STAT Computation/Homeworks/JRodoni_HW04_script.R
> # Created By: Jack Rodoni
> # Creation Date: 09/14/2021
> # Purpose: STAT 604 Homework 4
> # Last Executed: 09/14/2021
> Sys.time()
[1] "2021-09-14 11:10:27 CDT"
> 
> # 1.) Perform housekeeping steps to ensure you start with a clean workspace.The
> #     first housekeeping function should display the contents of the workspace.
> #     The second housekeeping function should clear the workspace but it is to
> #     be commented out so it will not be run automatically should you execute
> #     the entire script.  Add a step to show which libraries are loaded in your
> #     session.
> 
> ls()
character(0)
> rm(list = ls())
> search()
[1] ".GlobalEnv"        "package:stats"     "package:graphics"
[4] "package:grDevices" "package:utils"     "package:datasets"
[7] "package:methods"   "Autoloads"         "package:base"
> 
> 
> 
> # 2.) Use a function to set up your R session so that everything written to the
> #     console will also be directed to a separate text file while still
> #     appearing in the console.  Include the full path to show where the
> #     textfile will be written.
```

```
>
> sink(file = "C:/Users/jackr/OneDrive/Desktop/Graduate School Courses/STAT 604 -
STAT Computation/Homeworks/JRodoni_HW4.txt", split = TRUE)
>
>
> # 3.) Invoke R help to research the seq function in the available documentation.
> #     This command is not to be part of your program script but will be
> #     referenced as the answer to one of the questions at the end of the
> #     assignment.
>
>
> # 4.) Unless you are specifically instructed to give an object a certain name,
> #     you are expected to use a name of your own choosing.  Write a single
> #     line of code to create in the workspace and display a vector of numeric
> #     values from 5 to 80 with an increment of 5.  Show the type of data in the
> #     vector.  Show the length.
>
> (seq1 = seq(from = 5, to = 80, by = 5))
 [1]  5 10 15 20 25 30 35 40 45 50 55 60 65 70 75 80
> class(seq1)
[1] "numeric"
> length(seq1)
[1] 16
>
> # 5.) Create in the workspace and display a vector of numeric values from 0.4
> #     to 20 with an increment of 0.4.  Show the type of data inthe vector.
> #     Show the length.
>
> (seq2 = seq(from = 0.4, to = 20, by = 0.4))
 [1]  0.4  0.8  1.2  1.6  2.0  2.4  2.8  3.2  3.6  4.0  4.4  4.8  5.2  5.6  6.0
[16]  6.4  6.8  7.2  7.6  8.0  8.4  8.8  9.2  9.6 10.0 10.4 10.8 11.2 11.6 12.0
[31] 12.4 12.8 13.2 13.6 14.0 14.4 14.8 15.2 15.6 16.0 16.4 16.8 17.2 17.6 18.0
[46] 18.4 18.8 19.2 19.6 20.0
> class(seq2)
[1] "numeric"
> length(seq2)
[1] 50
>
> # 6.) Use the firstvector to create and display a matrix by columns that is 4
> #     columns wide
>
> (matrix1 = matrix(data = seq1, ncol = 4, byrow = FALSE))
     [,1] [,2] [,3] [,4]
[1,]    5   25   45   65
[2,]   10   30   50   70
[3,]   15   35   55   75
[4,]   20   40   60   80
>
> # 7.) Combine the two vectors as rows to create and display a new matrix.
>
```

```
> (matrix2 = rbind(seq1, seq2))
     [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12] [,13] [,14]
seq1  5.0 10.0 15.0 20.0   25 30.0 35.0 40.0 45.0    50  55.0  60.0  65.0  70.0
seq2  0.4  0.8  1.2  1.6    2  2.4  2.8  3.2  3.6     4   4.4   4.8   5.2   5.6
     [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22] [,23] [,24] [,25] [,26]
seq1    75  80.0   5.0  10.0  15.0    20  25.0  30.0  35.0  40.0    45  50.0
seq2     6   6.4   6.8   7.2   7.6     8   8.4   8.8   9.2   9.6    10  10.4
     [,27] [,28] [,29] [,30] [,31] [,32] [,33] [,34] [,35] [,36] [,37] [,38]
seq1  55.0  60.0  65.0    70  75.0  80.0   5.0  10.0    15  20.0  25.0  30.0
seq2  10.8  11.2  11.6    12  12.4  12.8  13.2  13.6    14  14.4  14.8  15.2
     [,39] [,40] [,41] [,42] [,43] [,44] [,45] [,46] [,47] [,48] [,49] [,50]
seq1  35.0    40  45.0  50.0  55.0  60.0    65  70.0  75.0  80.0   5.0    10
seq2  15.6    16  16.4  16.8  17.2  17.6    18  18.4  18.8  19.2  19.6    20
Warning message:
In rbind(seq1, seq2) :
  number of columns of result is not a multiple of vector length (arg 1)
>
> # 8.) Combine the two vectors as columns to create and display a new matrix.
>
> (matrix3 = cbind(seq1, seq2))
      seq1 seq2
 [1,]    5  0.4
 [2,]   10  0.8
 [3,]   15  1.2
 [4,]   20  1.6
 [5,]   25  2.0
 [6,]   30  2.4
 [7,]   35  2.8
 [8,]   40  3.2
 [9,]   45  3.6
[10,]   50  4.0
[11,]   55  4.4
[12,]   60  4.8
[13,]   65  5.2
[14,]   70  5.6
[15,]   75  6.0
[16,]   80  6.4
[17,]    5  6.8
[18,]   10  7.2
[19,]   15  7.6
[20,]   20  8.0
[21,]   25  8.4
[22,]   30  8.8
[23,]   35  9.2
[24,]   40  9.6
[25,]   45 10.0
[26,]   50 10.4
[27,]   55 10.8
[28,]   60 11.2
[29,]   65 11.6
```

```
[30,]   70 12.0
[31,]   75 12.4
[32,]   80 12.8
[33,]    5 13.2
[34,]   10 13.6
[35,]   15 14.0
[36,]   20 14.4
[37,]   25 14.8
[38,]   30 15.2
[39,]   35 15.6
[40,]   40 16.0
[41,]   45 16.4
[42,]   50 16.8
[43,]   55 17.2
[44,]   60 17.6
[45,]   65 18.0
[46,]   70 18.4
[47,]   75 18.8
[48,]   80 19.2
[49,]    5 19.6
[50,]   10 20.0
Warning message:
In cbind(seq1, seq2) :
  number of rows of result is not a multiple of vector length (arg 1)
>
> # 9.) Create a vector that contains the ninenumeric values 67, 72, 75,95,58,
> #     82, 88, 93 and 100. Execute a command that will display only the second,
> #     fourth,fifth and sixth members of the vector.
>
> vect1 = c(67, 72, 75, 95, 58, 82, 88, 93, 100)
> vect1[c(2,4,5,6)]
[1] 72 95 58 82
>
> # 10.) Create another vector that contains character strings with values of
> #      Dasher, Dancer, Prancer, Donder, Blitzen, Vixen, Comet, Cupid,and
> #      Rudolph. Execute a command that will display only the first four members
> #      of the vector.
>
> vect2 = c("Dasher", "Dancer", "Prancer", "Donder", "Blitzen", "Vixen", "Comet",
+           "Cupid", "Rudolph")
> vect2[1:4]
[1] "Dasher"  "Dancer"  "Prancer" "Donder"
>
> # 11.) Combine the character vector with the numeric vector to create and
> #      display a data frame.  Execute a function to show the data storage
> #      type of the new data frame.  Show the contents of the workspace
>
> (df1 = data.frame(vect1,vect2))
  vect1   vect2
1    67  Dasher
```

```
2    72  Dancer
3    75 Prancer
4    95  Donder
5    58 Blitzen
6    82   Vixen
7    88   Comet
8    93   Cupid
9   100 Rudolph
> mode(df1)
[1] "list"
> ls()
[1] "df1"     "matrix1" "matrix2" "matrix3" "seq1"    "seq2"    "vect1"
[8] "vect2"
>
>
> # 12.) Load the states workspace that you downloaded from Canvas. You may use
> #      the R menu to load the workspace initially,but your script must contain
> #      a line of code that will load the workspace the next time you run the
> #      script.  Some versions of R will make an entry in the console log
> #      showing the command that loaded the workspace.  If you get this line,
> #      you may copy it into your script.  Otherwise, you will need to find
> #      the command syntax inthe course slides orR documentation and write the
> #      command yourself. Show the contents of the workspace with the newly
> #      loaded object(s).
>
> load("C:/Users/jackr/OneDrive/Desktop/Graduate School Courses/STAT 604 - STAT
Computation/RData/states.RData")
> ls()
[1] "df1"     "matrix1" "matrix2" "matrix3" "seq1"    "seq2"    "Texas"
[8] "vect1"   "vect2"
>
> # 13.) Display the object type and the type of data in Texas.
> class(Texas) #object type
[1] "data.frame"
> mode(Texas)  #data type
[1] "list"
>
>
> # 14.) Display the object type and type of data in column 1 from Texas.
> class(Texas[,1]) #object type
[1] "factor"
> mode(Texas[,1])  # data type
[1] "numeric"
>
> # 15.) Display the structure of Texas
> str(Texas)
'data.frame':   254 obs. of  3 variables:
 $ CTYNAME   : Factor w/ 254 levels "Anderson County",..: 101 57 220 15 227 43 108
61 70 79 ...
 $ Pop       : int  4698619 2637772 2084931 1986049 1248743 1005146 865939 859064
```

```
840758 787858 ...
 $ GrowthRate: num  14.4 11.2 14.7 15.3 21.2 ...
>
> # 16.) Display a summary of Texas
> summary(Texas)
             CTYNAME          Pop             GrowthRate
 Anderson County :  1   Min.   :    152   Min.   :-18.596
 Andrews County  :  1   1st Qu.:   6968   1st Qu.: -2.532
 Angelina County :  1   Median :  18726   Median :  2.118
 Aransas County  :  1   Mean   : 112999   Mean   :  4.375
 Archer County   :  1   3rd Qu.:  52034   3rd Qu.:  8.930
 Armstrong County:  1   Max.   :4698619   Max.   : 80.952
 (Other)         :248
>
> # 17.) Display the first 20 rows and all but column 3 from Texas.Use a
> #      negative index value.
> Texas[1:20,-3]
             CTYNAME     Pop
1      Harris County 4698619
2      Dallas County 2637772
3     Tarrant County 2084931
4       Bexar County 1986049
5      Travis County 1248743
6      Collin County 1005146
7     Hidalgo County  865939
8      Denton County  859064
9     El Paso County  840758
10  Fort Bend County  787858
11 Montgomery County  590925
12 Williamson County  566719
13    Cameron County  423908
14   Brazoria County  370200
15     Nueces County  362265
16       Bell County  355642
17  Galveston County  337890
18    Lubbock County  307412
19       Webb County  275910
20  Jefferson County  255001
> # or alternatively
> Texas[-(21:nrow(Texas)),-3]
             CTYNAME     Pop
1      Harris County 4698619
2      Dallas County 2637772
3     Tarrant County 2084931
4       Bexar County 1986049
5      Travis County 1248743
6      Collin County 1005146
7     Hidalgo County  865939
8      Denton County  859064
9     El Paso County  840758
```

```
10  Fort Bend County  787858
11 Montgomery County  590925
12 Williamson County  566719
13    Cameron County  423908
14   Brazoria County  370200
15     Nueces County  362265
16       Bell County  355642
17  Galveston County  337890
18    Lubbock County  307412
19       Webb County  275910
20  Jefferson County  255001
>
> # 18.) Create and display a new object from Texas using the first 15 rows,
> #      the first column and third column
> (TexasB = Texas[1:15,c(1,3)])
             CTYNAME GrowthRate
1      Harris County    14.3821
2      Dallas County    11.1892
3     Tarrant County    14.7138
4       Bexar County    15.2766
5      Travis County    21.1738
6      Collin County    27.4869
7     Hidalgo County    11.1299
8      Denton County    28.8400
9     El Paso County     4.6168
10  Fort Bend County    33.4764
11 Montgomery County    28.6830
12 Williamson County    32.9440
13    Cameron County     3.9872
14   Brazoria County    17.7305
15     Nueces County     6.4693
>
> # 19.) Add a command that closes the text file and stops sending output to it.
> sink()
>
> # 20.) After you have run your script for the final time, answer the following
> #      questions in a series of comments at the bottom of the script.
>
> # a.) What command did you use to invoke help on seq?
> #       ?seq
>
> # b.) How many packages are loaded in your R Session?
> #     (Count only those listed as “package:”).
> #     7
>
> # c.) What type of data is inthe vector created in step 4?
> #     Numeric
>
> # d.) Explain how the values from the first vector are used
> #     in the creation of the matrix in step 7.
```

```
>
> #     The values of the first vector are used as the values in
> #     the first row of the matrix. It is important to note that
> #     becasue the length of vector 2 is greater than the length
> #     of vector 1, the values of vector 1 are recycled inorder to
> #     make both rows of the matrix the same length.
>
> # e.) What is the type of data in the data frame created in step 11?
> #     list
>
> # f.) What is the class and data type of column 1 from Texas?
> #     Class - factor
> #     Data Type - numeric
>
> # g.) How many observations and variables are in the Texas data frame?
> #     254 observations of 3 variables
>
> # h.) Explain the relationship between the median and mean of the Pop column?
> #     The mean pop is much higher than the median pop as the mean is skewed
> #     heavily by the high population counties.
>
>
```