CPSC 260
Spring 2022

Instructions: This assignment will give you a chance to practice what you have learned while working through projects #1 (dice) and #2 (cards), specifically chapters 1-4. For each problem copy and paste your R code from the R script file window AND your output from the Console window into this Word document. [You should always run your code and view the results, even if this is not specified in the problem!] Please use a color other than black. You will upload this Word document to BB when finished.

Use Beginning R by Gardener as a resource. It provides many more examples and explanations of the concepts we have covered. Below I have listed blocks of that text that may be helpful.

- Reading Bigger Data files [pg 37] through Alternative Commands for Reading Data in R
- Types of Data Items [pg 45] through Examining Data Structure
- Writing Matrix and Data Frame Objects to Disk [pg 58]
- Manipulating Vectors [pg 66] through Selecting and Displaying Parts of a Matrix or Data Frame
- Viewing and Setting Names [pg 82]
- Making Data Frames [pg 88] through Making Matrix Objects
- Converting from One Object Form to Another [pg 97] through Convert a Data Frame into a Matrix

1. You have the results of a simple experiment to look at the visitation of various bee species to 5 different plants. The number of bees observed was as follows:

Buff tail: 10 1 37 5 12
Garden bee: 8 3 19 6 4
Red tail: 18 9 1 2 4
Honeybee: 12, 13, 16, 9, 10
Carder bee: 8, 27, 6, 32, 23

a) Make numeric a numeric vector, containing the above data, for each type of bee. Make sure you give each vector an informative name.

```
buff_tail<-c(10,1,37,5,12)
garden_bee<-c(8,3,19,6,4)
red_tail<-c(18,9,1,2,4)
honeybee<-c(12,13,16,9,10)
carder_bee<-c(8,27,6,32,23)

bees<-c("Buff Tail", buff_tail,"Garden bee", garden_bee,"Red tail", red_tail,"Honeybee",
honeybee,"Carder bee", carder_bee)

bees
```

```
[1] "Buff Tail"  "10"         "1"
 [4] "37"          "5"           "12"
 [7] "Garden bee" "8"           "3"
[10] "19"          "6"           "4"
[13] "Red tail"    "18"          "9"
[16] "1"           "2"           "4"
[19] "Honeybee"    "12"          "13"
[22] "16"          "9"           "10"
```

```
[25] "Carder bee" "8"          "27"
[28] "6"          "32"         "23"
```

b) Make a matrix object from the bee vectors in such a way that each vector becomes a column in the matrix. Make sure you give the resulting matrix an informative name.

```
bee.data<-c(buff_tail,garden_bee,red_tail,honeybee,carder_bee)
bee_matrix <- matrix(bee.data,nrow=5,ncol=5)
colnames(bee_matrix) <- c("Buff tail", "Garden bee", "Red tail", "Honeybee", "Carder bee")
bee_matrix
```

```
Buff tail Garden bee Red tail Honeybee
[1,]        10          8       18       12
[2,]         1          3        9       13
[3,]        37         19        1       16
[4,]         5          6        2        9
[5,]        12          4        4       10
     Carder bee
[1,]          8
[2,]         27
[3,]          6
[4,]         32
[5,]         23
```

c) Convert your matrix into a data frame. Make sure you store the data frame in a new object that has an informative name.

```
bee_data_frame = data.frame(bee_matrix)
bee_data_frame
```

```
  Buff.tail Garden.bee Red.tail Honeybee
1        10          8       18       12
2         1          3        9       13
3        37         19        1       16
4         5          6        2        9
5        12          4        4       10
  Carder.bee
1          8
2         27
3          6
4         32
5         23
```

d) Save the data frame to your computer using either the *write.table() or write.csv()* function.

```
write.table(bee_data_frame)
```

```
"Buff.tail" "Garden.bee" "Red.tail" "Honeybee" "Carder.bee"
"1" 10 8 18 12 8
"2" 1 3 9 13 27
"3" 37 19 1 16 6
"4" 5 6 2 9 32
"5" 12 4 4 10 23
```

2. Refer to problem #1. You will continue using the bee vectors that you created.

a) Make a data frame object directly from the bee vectors. In other words, store all vectors together in a single data frame object. Make sure you give the resulting data frame an informative name.

```
bee_data_frame2 = data.frame(buff_tail, garden_bee, red_tail, honeybee, carder_bee)
bee_data_frame2

  buff_tail garden_bee red_tail honeybee
1        10          8       18       12
2         1          3        9       13
3        37         19        1       16
4         5          6        2        9
5        12          4        4       10
  carder_bee
1          8
2         27
3          6
4         32
5         23
```

b) Convert your new data frame into a matrix. Make sure you store the matrix in a new object that has an informative name.

```
bee_matrix2 = as.matrix(bee_data_frame2)
> class(bee_matrix2)
[1] "matrix"

bee_matrix2
     buff_tail garden_bee red_tail honeybee
[1,]        10          8       18       12
[2,]         1          3        9       13
[3,]        37         19        1       16
[4,]         5          6        2        9
[5,]        12          4        4       10
     carder_bee
[1,]          8
[2,]         27
[3,]          6
[4,]         32
[5,]         23
```

c) Each row of your matrix relates to a specific plant. So you could assign names to the rows. The plants, in order, are: Thistle, Vipers bugloss, Golden rain, Yellow alfalfa, and Blackberry. Create a vector containing the plant names.

```
plants = c("Thistle", "Vipers_bugloss", "Golden_rain", "Ellow_alfalfa", "Blackberry")
plants

[1] "Thistle"        "Vipers_bugloss"
[3] "Golden_rain"    "Ellow_alfalfa"
[5] "Blackberry"
```

d) Use the vector of plant names that you created to add row labels to your matrix. You may need to read the help file for the function *row.names()* to accomplish this.

```
row.names(bee_matrix2) = c(plants)
bee_matrix2
```

```
               buff_tail garden_bee red_tail
Thistle               10          8       18
Vipers_bugloss         1          3        9
Golden_rain           37         19        1
Ellow_alfalfa          5          6        2
Blackberry            12          4        4
               honeybee carder_bee
Thistle              12          8
Vipers_bugloss       13         27
Golden_rain          16          6
Ellow_alfalfa         9         32
Blackberry           10         23
```

3. Use the data frame from either 1c or 2a. Write the necessary code to display only the specified portions of the data frame. Use both the square bracket notation [row, col] and dollar sign ($) notation when applicable.

a) data for Blackberry only

```
row.names(bee_data_frame2) = plants
bee_data_frame2

bee_data_frame2["Blackberry", ]
```

```
buff_tail garden_bee red_tail
Blackberry        12           4        4
           honeybee carder_bee
Blackberry       10          23
```

b) data for just Golden rain and Yellow alfalfa

```
bee_data_frame2[3:4, ]
```

```
               buff_tail garden_bee red_tail
Golden_rain           37         19        1
Yellow_alfalfa         5          6        2
               honeybee carder_bee
Golden_rain          16          6
Yellow_alfalfa        9         32
```

c) data for just Golden rain and the Buff tail

```
bee_data_frame2[3, 1]
```

```
[1] 37
```

d) data for the Red tail bee only

```
bee_data_frame2[ , 3]
[1] 18  9  1  2  4
> bee_data_frame2$red_tail
[1] 18  9  1  2  4
```

e) data for all bees, except the Red tail

```
bee_data_frame2[1:5, -3]

               buff_tail garden_bee honeybee
Thistle               10          8       12
Vipers_bugloss         1          3       13
Golden_rain           37         19       16
Yellow_alfalfa         5          6        9
Blackberry            12          4       10
               carder_bee
Thistle                 8
Vipers_bugloss         27
Golden_rain             6
Yellow_alfalfa         32
Blackberry             23
```

f) data for just the plants where at least 10 Honeybees were observed

```
bee_data_frame2[-4,4]
[1] 12 13 16 10
```

g) data for just the plants where at least 10 Honeybees and Buff tails were observed

```
filter(bee_data_frame2, !honeybee %in% c(9) & !buff_tail %in% c(1,5))

            buff_tail garden_bee red_tail honeybee carder_bee
Thistle            10          8       18       12          8
Golden_rain        37         19        1       16          6
Blackberry         12          4        4       10         23
```

4. Refer to the card deck and the shuffle() and deal() functions that you created in-class.

a) Load your card deck into R. In other words use either the *read.table()* or *read.csv()* function to bring the dataset *deck.csv* (posted to Bb) into R.

```
NewDeck<-read.csv(file=file.choose())
```

b) Create a new deal() function. This one should deal out the top 5 cards (e.g. 5 card poker hand). Make sure to give your function an informative name. Use your function to deal out several hands. Shuffle the deck between deals.

```
newdeal <- function() {
  card <- deck[1:5, ]
  assign("deck", deck[-1 & -2 & -3 & -4 & -5 ], envir = globalenv())
  card
}
newshuffle <- function(){
  random <- sample(1:52, size = 52)
  assign("deck", DECK[random, ], envir = globalenv())
}

newshuffle()
newdeal()
```

```
newshuffle()
> newdeal()
    face     suit value
4    ten   spades    10
24 three    clubs     3
30   ten diamonds    10
NA  <NA>     <NA>    NA
25   two    clubs     2
> newshuffle()
> newdeal()
    face   suit value
5   nine spades     9
16  jack  clubs    11
15 queen  clubs    12
40  king hearts    13
26   ace  clubs     1
```

c) Create another new deal() function. This one should deal out 5 cards to 3 players, mimicking how cards are normally dealt [one at a time, going around in a circle]. If you don’t know what I mean, then please watch this video - starting around 0:50:

https://www.youtube.com/watch?v=UoSHyGyVAYM

The function should return a list object, with each element in the list being one of the player’s hands. Do NOT include the point values with the cards – just the face and suit of each. Make sure to give your function an informative name. Test your function multiple times. Shuffle the deck between deals.

```
shuffle5c = shuffle(deck){
  shuffle = sample(1:52, size = 52)
  deck[shuffle, ]
}

shuffle5d = shuffle5c(deck)

deal5<-function(){
  shuffle5d = shuffle5d[c(1), ]
  head(shuffle5d, n=5)
}
```

```
  p1 = deal5()
  p2 = deal5()
  p3 = deal5()

error
```