R Notebook

Name: TEAN JIN HE MatricID: SD21063

Exercise 1

Question a

Hide

```
a <- 2.3
(6*a+42)/3^(4.2-3.62)
```

```
[1] 29.50556
```

Question b

Hide

```
log(12.43)
```

```
[1] 2.520113
```

Hide

```
log10(12.43) #log to base 10
```

```
[1] 1.094471
```

Hide

```
log2(12.43) #log to base 2
```

```
[1] 3.635754
```

Hide

```
log(x=12.43, base=2) # alternative log to base 2
```

```
[1] 3.635754
```

Hide

```
sqrt(12.43)
```

```
[1] 3.525621
```

```
exp(12.43)
```

```
[1] 250196
```

Question c

```
d <- 20
r <- d/2
area_circle <- pi*(r^2)
area_circle
```

```
[1] 314.1593
```

Question d

```
weight <- c(69,63,57,55,47,48,66,50,49,45)
weight_sort = sort(weight, decreasing = FALSE)
weight_sort[weight_sort<50]
```

```
[1] 45 47 48 49
```

Exercise 2

Question a

```
v1 <- seq(5,11, length=7);v1
```

```
[1]   5   6   7   8   9 10 11
```

```
v2 <- c(3,-1,4.5,-37)
rep(v2,times=2)
```

```
[1]   3.0  -1.0   4.5 -37.0   3.0  -1.0   4.5 -37.0
```

```
v3 <- sqrt(4+(2/3));v3
```

```
[1] 2.160247
```

```
v4 <- c(v1,v2,v3);v4
```

```
 [1]    5.000000    6.000000    7.000000    8.000000    9.000000   10.000000
 [7]   11.000000    3.000000   -1.000000    4.500000  -37.000000    2.160247
```

Question b

```
v5 <- v4[c(5,8,12)];v5
```

```
[1] 9.000000 3.000000 2.160247
```

Question c

```
length(v4)
```

```
[1] 12
```

```
x <- v4[-c(1,12)]
x
```

```
 [1]    6.0    7.0    8.0    9.0   10.0   11.0    3.0   -1.0    4.5  -37.0
```

Question d

```
v6 = sort(v4, decreasing = TRUE);v6
```

```
 [1]   11.000000   10.000000    9.000000    8.000000    7.000000    6.000000
 [7]    5.000000    4.500000    3.000000    2.160247   -1.000000  -37.000000
```

Question e

```
x1 <- append(x,rep(7,times=2),after=2)
x2 <- append(x1,rep(10,times=4),after=7)
x3 <- append(x2,rep(-37,times=1),after=16)
x3
```

```
 [1]   6.0   7.0   7.0   7.0   8.0   9.0  10.0  10.0  10.0  10.0  10.0
[12]  11.0   3.0  -1.0   4.5 -37.0 -37.0
```

Exercise 3

Question a

Hide

```
##file.choose()

dt <- read.csv("C:/Users/user/OneDrive/Desktop/Sem4 slide/data science programming 2/weight_heig
ht_BSD2223_asal2.csv", header = TRUE)
dt
```

Timestamp <chr>	Section <chr>	ID_no <int>	Gender <chr>	weight <chr>	height <chr>
2023/03/22 5:49:05 AM GMT+8	Section 02G	1001		50	1.7
2023/03/22 6:01:52 AM GMT+8	Section 01G	1002	Female	72	1.66
2023/03/22 6:02:15 AM GMT+8	Section 02G	1003	Female	55kg	160
2023/03/22 6:22:17 AM GMT+8	Section 01G	1004	Female	53.9	1.56
2023/03/22 6:33:07 AM GMT+8	Section 01G	1005	Female	72	1.6
2023/03/22 6:44:03 AM GMT+8	Section 01G	1006	Female	78	1.56
2023/03/22 6:47:38 AM GMT+8	Section 02G	1007	Male	71	179
2023/03/22 6:48:52 AM GMT+8	Section 01G	1008	Female	46	161
2023/03/22 6:53:59 AM GMT+8	Section 01G	1009	Male	55	184
2023/03/22 6:55:58 AM GMT+8	Section 01G	1010	Female	57	165

1-10 of 71 rows   Previous **1** 2 3 4 5 6 … 8 Next

Question b

Hide

```
str(dt)
```

```
'data.frame':   71 obs. of  6 variables:
 $ Timestamp: chr  "2023/03/22 5:49:05 AM GMT+8" "2023/03/22 6:01:52 AM GMT+8" "2023/03/22 6:02:
15 AM GMT+8" "2023/03/22 6:22:17 AM GMT+8" ...
 $ Section  : chr  "Section 02G" "Section 01G" "Section 02G" "Section 01G" ...
 $ ID_no    : int  1001 1002 1003 1004 1005 1006 1007 1008 1009 1010 ...
 $ Gender   : chr  "" "Female" "Female" "Female" ...
 $ weight   : chr  "50" "72" "55kg" "53.9" ...
 $ height   : chr  "1.7" "1.66" "160" "1.56" ...
```

Question c

```
head(dt,10)
```

	Timestamp <chr>	Section <chr>	ID_no <int>	Gender <chr>	weight <chr>	height <chr>
1	2023/03/22 5:49:05 AM GMT+8	Section 02G	1001		50	1.7
2	2023/03/22 6:01:52 AM GMT+8	Section 01G	1002	Female	72	1.66
3	2023/03/22 6:02:15 AM GMT+8	Section 02G	1003	Female	55kg	160
4	2023/03/22 6:22:17 AM GMT+8	Section 01G	1004	Female	53.9	1.56
5	2023/03/22 6:33:07 AM GMT+8	Section 01G	1005	Female	72	1.6
6	2023/03/22 6:44:03 AM GMT+8	Section 01G	1006	Female	78	1.56
7	2023/03/22 6:47:38 AM GMT+8	Section 02G	1007	Male	71	179
8	2023/03/22 6:48:52 AM GMT+8	Section 01G	1008	Female	46	161
9	2023/03/22 6:53:59 AM GMT+8	Section 01G	1009	Male	55	184
10	2023/03/22 6:55:58 AM GMT+8	Section 01G	1010	Female	57	165

1-10 of 10 rows

```
tail(dt,10)
```

	Timestamp <chr>	Section <chr>	ID_no <int>	Gender <chr>	weight <chr>	height <chr>
62	2023/03/23 7:59:15 AM GMT+8	Section 03G	1062	Female	65	1.53
63	2023/03/23 7:59:22 AM GMT+8	Section 03G	1063	Female	60	1.5
64	2023/03/23 8:07:41 AM GMT+8	Section 03G	1064	Male	73	170
65	2023/03/23 8:10:13 AM GMT+8	Section 03G	1065	Male	96	165
66	2023/03/23 8:38:44 AM GMT+8	Section 03G	1066	Female	45	162
67	2023/03/23 8:38:54 AM GMT+8	Section 03G	1067	Male	54	165
68	2023/03/23 8:39:36 AM GMT+8	Section 03G	1068	Male	60	175
69	2023/03/23 8:40:07 AM GMT+8	Section 03G	1069	Female	60	1.63
70	2023/03/23 8:40:35 AM GMT+8	Section 03G	1070	Female	43	152
71	2023/03/23 8:41:51 AM GMT+8	Section 03G	1071	Female	50	1.64

Question d

Hide

```
dt1 <- dt[,-1]
dt1
```

Section <chr>	ID_no <int>	Gender <chr>	weight <chr>	height <chr>
Section 02G	1001		50	1.7
Section 01G	1002	Female	72	1.66
Section 02G	1003	Female	55kg	160
Section 01G	1004	Female	53.9	1.56
Section 01G	1005	Female	72	1.6
Section 01G	1006	Female	78	1.56
Section 02G	1007	Male	71	179
Section 01G	1008	Female	46	161
Section 01G	1009	Male	55	184
Section 01G	1010	Female	57	165

1-10 of 71 rows   Previous **1** 2 3 4 5 6 … 8 Next

Question e

Hide

```
class(dt1[,c(4)])
```

```
[1] "character"
```

Hide

```
class(dt1[,c(5)])
```

```
[1] "character"
```

Hide

```
transform(dt1,weight = as.numeric(weight))
```

```
Warning: NAs introduced by coercion
```

Section	ID_no	Gender	weight	height
<chr>	<int>	<chr>	<dbl>	<chr>
Section 02G	1001		50.0	1.7
Section 01G	1002	Female	72.0	1.66
Section 02G	1003	Female	NA	160
Section 01G	1004	Female	53.9	1.56
Section 01G	1005	Female	72.0	1.6
Section 01G	1006	Female	78.0	1.56
Section 02G	1007	Male	71.0	179
Section 01G	1008	Female	46.0	161
Section 01G	1009	Male	55.0	184
Section 01G	1010	Female	57.0	165

1-10 of 71 rows   Previous **1** 2 3 4 5 6 … 8 Next

Hide

```
class(dt1[,c(4)])
```

```
[1] "character"
```

Hide

```
# change the character columns to numeric
dt1$weight = as.numeric(as.character(dt1$weight))
```

```
Warning: NAs introduced by coercion
```

Hide

```
dt1$height = as.numeric(as.character(dt1$height))
```

```
Warning: NAs introduced by coercion
```

Hide

```
print(sapply(dt1, class))
```

```
    Section        ID_no       Gender      weight       height
 "character"    "integer"  "character"    "numeric"    "numeric"
```

```
colSums(is.na(dt1))
```

```
Section    ID_no   Gender   weight   height
     0        0        0        2        1
```

```
dt1[3, 4] <- 55
dt1[59, 4] <- 55
dt1[32, 5] <- 1.77
dt1
```

Section	ID_no	Gender	weight	height
<chr>	<int>	<chr>	<dbl>	<dbl>
Section 02G	1001		50.0	1.70
Section 01G	1002	Female	72.0	1.66
Section 02G	1003	Female	55.0	160.00
Section 01G	1004	Female	53.9	1.56
Section 01G	1005	Female	72.0	1.60
Section 01G	1006	Female	78.0	1.56
Section 02G	1007	Male	71.0	179.00
Section 01G	1008	Female	46.0	161.00
Section 01G	1009	Male	55.0	184.00
Section 01G	1010	Female	57.0	165.00

1-10 of 71 rows   Previous **1** 2 3 4 5 6 … 8 Next

```
library(dplyr)
# if else condition of dataframe column  in R
dt2 <- mutate(dt1, height = ifelse(height >= 100, height/100, height))
dt2
```

Section	ID_no	Gender	weight	height
<chr>	<int>	<chr>	<dbl>	<dbl>
Section 02G	1001		50.0	1.70

Section	ID_no	Gender	weight	height
<chr>	<int>	<chr>	<dbl>	<dbl>
Section 01G	1002	Female	72.0	1.66
Section 02G	1003	Female	55.0	1.60
Section 01G	1004	Female	53.9	1.56
Section 01G	1005	Female	72.0	1.60
Section 01G	1006	Female	78.0	1.56
Section 02G	1007	Male	71.0	1.79
Section 01G	1008	Female	46.0	1.61
Section 01G	1009	Male	55.0	1.84
Section 01G	1010	Female	57.0	1.65

1-10 of 71 rows   Previous **1** 2 3 4 5 6 … 8 Next

Question f

Hide

```
# calculate BMI value

dt3 <- dt2 %>% mutate(Section, bmi = weight / (height ^ 2)) %>% select(Section:weight, height, bmi)
dt3
```

Section	ID_no	Gender	weight	height	bmi
<chr>	<int>	<chr>	<dbl>	<dbl>	<dbl>
Section 02G	1001		50.0	1.70	17.30104
Section 01G	1002	Female	72.0	1.66	26.12861
Section 02G	1003	Female	55.0	1.60	21.48437
Section 01G	1004	Female	53.9	1.56	22.14826
Section 01G	1005	Female	72.0	1.60	28.12500
Section 01G	1006	Female	78.0	1.56	32.05128
Section 02G	1007	Male	71.0	1.79	22.15911
Section 01G	1008	Female	46.0	1.61	17.74623
Section 01G	1009	Male	55.0	1.84	16.24527
Section 01G	1010	Female	57.0	1.65	20.93664

1-10 of 71 rows   Previous **1** 2 3 4 5 6 … 8 Next

Question g

```
# multiple if else condition of dataframe column   in R

dt_clean <- mutate(dt3, categories = ifelse(bmi < 18.5, "Underweight",
                            ifelse(bmi < 25, "Normal",
                                 ifelse(bmi < 30, "Overweight",
                                      ifelse(bmi < 35, "Obese", "Extremely Obese")))))
dt_clean
```

Section <chr>	ID_no <int>	Gender <chr>	weight <dbl>	height <dbl>	bmi <dbl>	categories <chr>
Section 02G	1001		50.0	1.70	17.30104	Underweight
Section 01G	1002	Female	72.0	1.66	26.12861	Overweight
Section 02G	1003	Female	55.0	1.60	21.48437	Normal
Section 01G	1004	Female	53.9	1.56	22.14826	Normal
Section 01G	1005	Female	72.0	1.60	28.12500	Overweight
Section 01G	1006	Female	78.0	1.56	32.05128	Obese
Section 02G	1007	Male	71.0	1.79	22.15911	Normal
Section 01G	1008	Female	46.0	1.61	17.74623	Underweight
Section 01G	1009	Male	55.0	1.84	16.24527	Underweight
Section 01G	1010	Female	57.0	1.65	20.93664	Normal

1-10 of 71 rows   Previous **1** 2 3 4 5 6 … 8 Next

Exercise 4

Question a

```
C_degree <- function(F) (5/9)*(F-32)
C_degree
```

```
function(F) (5/9)*(F-32)
```

Question b

```
F <- c(47,80,25,20,105,132,30,274,33,214,58,77)

vC <- sapply(F, C_degree)
round(vC, 2)
```

```
[1]   8.33  26.67  -3.89  -6.67  40.56  55.56  -1.11 134.44   0.56
[10] 101.11  14.44  25.00
```

Exercise 5

```
A <- c(2,1,-3)
B <- c(1,4,0,3,2,8)
C <- c(1,4,0,3,2,8,-1,5,0)
MatrixA <- matrix(A, nrow = 3, ncol = 1); MatrixA
```

```
      [,1]
[1,]    2
[2,]    1
[3,]   -3
```

```
MatrixB <- matrix(B, nrow = 3, ncol = 2); MatrixB
```

```
     [,1] [,2]
[1,]    1    3
[2,]    4    2
[3,]    0    8
```

```
MatrixC <- matrix(C, nrow = 3, ncol = 3); MatrixC
```

```
     [,1] [,2] [,3]
[1,]    1    3   -1
[2,]    4    2    5
[3,]    0    8    0
```

Question a

```
# 5a (3x1) != (3x2)
MatrixA %*% MatrixB
```

```
Error in MatrixA %*% MatrixB : non-conformable arguments
```

Question b

```
t(MatrixA) %*% MatrixB
```

```
      [,1] [,2]
[1,]    6  -16
```

Question c

```
t(MatrixB) %*% (MatrixA %*% t(MatrixA))
```

```
      [,1] [,2] [,3]
[1,]    12    6  -18
[2,]   -32  -16   48
```

Question d

```
(MatrixA %*% t(MatrixA)) %*% t(MatrixB)
```

```
Error in (MatrixA %*% t(MatrixA)) %*% t(MatrixB) :
  non-conformable arguments
```

```
# 5e
((MatrixB %*% t(MatrixB))+(MatrixA %*% t(MatrixA))-(10/3))**-1
```

```
           [,1]       [,2]       [,3]
[1,] 0.09375000 0.11538462 0.06818182
[2,] 0.11538462 0.05660377 0.10344828
[3,] 0.06818182 0.10344828 0.01435407
```

Exercise 6

Question a

```
v1 <- seq(5.5,0.5)
Array1 <- array(v1, dim = c(4,2,6)); Array1
```

```
, , 1

      [,1] [,2]
[1,]  5.5  1.5
[2,]  4.5  0.5
[3,]  3.5  5.5
[4,]  2.5  4.5

, , 2

      [,1] [,2]
[1,]  3.5  5.5
[2,]  2.5  4.5
[3,]  1.5  3.5
[4,]  0.5  2.5

, , 3

      [,1] [,2]
[1,]  1.5  3.5
[2,]  0.5  2.5
[3,]  5.5  1.5
[4,]  4.5  0.5

, , 4

      [,1] [,2]
[1,]  5.5  1.5
[2,]  4.5  0.5
[3,]  3.5  5.5
[4,]  2.5  4.5

, , 5

      [,1] [,2]
[1,]  3.5  5.5
[2,]  2.5  4.5
[3,]  1.5  3.5
[4,]  0.5  2.5

, , 6

      [,1] [,2]
[1,]  1.5  3.5
[2,]  0.5  2.5
[3,]  5.5  1.5
[4,]  4.5  0.5
```

Question b

Hide

```
Array1[3,2,4]
```

```
[1] 5.5
```

Question c

```
Array1[2,,1]
```

```
[1] 4.5 0.5
```

Exercise 7

Question b

```
load(file="dt_clean.RData")
dt1_clean
```

Timestamp <chr>	Section <chr>	ID_no <int>	Gender <chr>	weight <dbl>	height <dbl>
2023/03/22 5:49:05 AM GMT+8	Section 02G	1001	NA	50.0	1.70
2023/03/22 6:01:52 AM GMT+8	Section 01G	1002	Female	72.0	1.66
2023/03/22 6:02:15 AM GMT+8	Section 02G	1003	Female	55.0	1.60
2023/03/22 6:22:17 AM GMT+8	Section 01G	1004	Female	53.9	1.56
2023/03/22 6:33:07 AM GMT+8	Section 01G	1005	Female	72.0	1.60
2023/03/22 6:44:03 AM GMT+8	Section 01G	1006	Female	78.0	1.56
2023/03/22 6:47:38 AM GMT+8	Section 02G	1007	Male	71.0	1.79
2023/03/22 6:48:52 AM GMT+8	Section 01G	1008	Female	46.0	1.61
2023/03/22 6:53:59 AM GMT+8	Section 01G	1009	Male	55.0	1.84
2023/03/22 6:55:58 AM GMT+8	Section 01G	1010	Female	57.0	1.65

1-10 of 71 rows   Previous **1** 2 3 4 5 6 … 8 Next

Question c

```
dt1_clean$height
```

```
 [1] 1.70 1.66 1.60 1.56 1.60 1.56 1.79 1.61 1.84 1.65 1.53 1.56 1.63
[14] 1.64 1.78 1.55 1.65 1.75 1.78 1.78 1.57 1.52 1.76 1.73 1.55 1.48
[27] 1.71 1.65 1.59 1.59 1.54 1.77 1.71 1.60 1.53 1.56 1.60 1.56 1.60
[40] 1.55 1.52 1.68 1.75 1.56 1.68 1.70 1.55 1.50 1.68 1.55 1.59 1.72
[53] 1.79 1.73 1.50 1.76 1.52 1.57 1.55 1.58 1.63 1.53 1.50 1.70 1.65
[66] 1.62 1.65 1.75 1.63 1.52 1.64
```

Question d

Hide

```
## by default right value inclusive

lbs <- c("height<1.5", "1.5<height<1.6", "height>1.6")
height_group <- cut(dt1_clean$height, breaks = c(0,1.5,1.6,Inf),labels=lbs)
height_group
```

```
 [1] height>1.6      height>1.6      1.5<height<1.6 1.5<height<1.6
 [5] 1.5<height<1.6 1.5<height<1.6 height>1.6      height>1.6
 [9] height>1.6      height>1.6      1.5<height<1.6 1.5<height<1.6
[13] height>1.6      height>1.6      height>1.6      1.5<height<1.6
[17] height>1.6      height>1.6      height>1.6      height>1.6
[21] 1.5<height<1.6 1.5<height<1.6 height>1.6      height>1.6
[25] 1.5<height<1.6 height<1.5      height>1.6      height>1.6
[29] 1.5<height<1.6 1.5<height<1.6 1.5<height<1.6 height>1.6
[33] height>1.6      1.5<height<1.6 1.5<height<1.6 1.5<height<1.6
[37] 1.5<height<1.6 1.5<height<1.6 1.5<height<1.6 1.5<height<1.6
[41] 1.5<height<1.6 height>1.6      height>1.6      1.5<height<1.6
[45] height>1.6      height>1.6      1.5<height<1.6 height<1.5
[49] height>1.6      1.5<height<1.6 1.5<height<1.6 height>1.6
[53] height>1.6      height>1.6      height<1.5      height>1.6
[57] 1.5<height<1.6 1.5<height<1.6 1.5<height<1.6 1.5<height<1.6
[61] height>1.6      1.5<height<1.6 height<1.5      height>1.6
[65] height>1.6      height>1.6      height>1.6      height>1.6
[69] height>1.6      1.5<height<1.6 height>1.6
Levels: height<1.5 1.5<height<1.6 height>1.6
```

Hide

```
levels(height_group) ## number of group
```

```
[1] "height<1.5"     "1.5<height<1.6" "height>1.6"
```

Hide

```
## combine weight data with group
cbind(dt1_clean$height,height_group)
```

```
              height_group
 [1,] 1.70              3
 [2,] 1.66              3
 [3,] 1.60              2
 [4,] 1.56              2
 [5,] 1.60              2
 [6,] 1.56              2
 [7,] 1.79              3
 [8,] 1.61              3
 [9,] 1.84              3
[10,] 1.65              3
[11,] 1.53              2
[12,] 1.56              2
[13,] 1.63              3
[14,] 1.64              3
[15,] 1.78              3
[16,] 1.55              2
[17,] 1.65              3
[18,] 1.75              3
[19,] 1.78              3
[20,] 1.78              3
[21,] 1.57              2
[22,] 1.52              2
[23,] 1.76              3
[24,] 1.73              3
[25,] 1.55              2
[26,] 1.48              1
[27,] 1.71              3
[28,] 1.65              3
[29,] 1.59              2
[30,] 1.59              2
[31,] 1.54              2
[32,] 1.77              3
[33,] 1.71              3
[34,] 1.60              2
[35,] 1.53              2
[36,] 1.56              2
[37,] 1.60              2
[38,] 1.56              2
[39,] 1.60              2
[40,] 1.55              2
[41,] 1.52              2
[42,] 1.68              3
[43,] 1.75              3
[44,] 1.56              2
[45,] 1.68              3
[46,] 1.70              3
[47,] 1.55              2
[48,] 1.50              1
[49,] 1.68              3
[50,] 1.55              2
[51,] 1.59              2
```

```
[52,] 1.72            3
[53,] 1.79            3
[54,] 1.73            3
[55,] 1.50            1
[56,] 1.76            3
[57,] 1.52            2
[58,] 1.57            2
[59,] 1.55            2
[60,] 1.58            2
[61,] 1.63            3
[62,] 1.53            2
[63,] 1.50            1
[64,] 1.70            3
[65,] 1.65            3
[66,] 1.62            3
[67,] 1.65            3
[68,] 1.75            3
[69,] 1.63            3
[70,] 1.52            2
[71,] 1.64            3
```

Question e

Hide

```
## frequency in each group
table(height_group)
```

```
height_group
    height<1.5 1.5<height<1.6      height>1.6
            4               31              36
```

Hide

```
#OR

## Using factor
f_height_group <- factor(height_group, levels=c("height<1.5", "1.5<height<1.6", "height>1.6"),or
dered=TRUE)
table(f_height_group)
```

```
f_height_group
    height<1.5 1.5<height<1.6      height>1.6
            4               31              36
```

Hide

```
#freq. table of proportions for height variable
prop.table(table(height_group))
```

```
height_group
    height<1.5 1.5<height<1.6     height>1.6
    0.05633803     0.43661972     0.50704225
```

Question f

<div style="text-align: right;">Hide</div>

```
## 2-way freq. table (by Section)
table(dt1_clean$Section, height_group)
```

```
          height_group
           height<1.5 1.5<height<1.6 height>1.6
  Section 01G          0             14         16
  Section 02G          2             13          8
  Section 03G          2              4         12
```

<div style="text-align: right;">Hide</div>

```
##2-way freq. table (by gender)
table(dt1_clean$Gender, height_group)
```

```
        height_group
         height<1.5 1.5<height<1.6 height>1.6
  Female          4             30         12
  Male            0              1         23
```