SHETH L.U.J.AND SIR M.V. COLLEGE
PRACTICAL(1,2,3,4,5,6) MODULE2
Data Analysis with SAS / SPSS /R

1) AIM:- Generating descriptive statistics using summary() or describe() (R)

INPUT:-

```
library(dplyr)

df <- read.csv("shades.csv")

print("---- 2. Frequency Tables for Shades Dataset ----")

# Automatically select the first categorical column

cat_var <- names(df)[!sapply(df, is.numeric)][1]


print(paste("Using categorical column:", cat_var))


print("Frequency Table using table():")

print(table(df[[cat_var]]))


print("Frequency Table using count():")

print(df %>% count(.data[[cat_var]]))
```

OUTPUT:-

```
Console   Terminal   Background Jobs
R 4.1.2 · ~/
> shades <- read.csv("C:/Users/mvlui/Downloads/shades.csv")
>   View(shades)
> library(dplyr)
>
> df <- read.csv("shades.csv")
>
> print("---- 2. Frequency Tables for Shades Dataset ----")
[1] "---- 2. Frequency Tables for Shades Dataset ----"
>
> # Automatically select the first categorical column
> cat_var <- names(df)[!sapply(df, is.numeric)][1]
>
> print(paste("Using categorical column:", cat_var))
[1] "Using categorical column: brand"
>
> print("Frequency Table using table():")
[1] "Frequency Table using table():"
> print(table(df[[cat_var]]))

       Addiction     bareMinerals   Beauty Bakerie   Bharat & Doris      Black Opal        Black Up     Blue Heaven     Bobbi Brown
              17               29               30                7              12              18               2              30
         Colorbar Covergirl + Olay             Dior       Elsas Pro  EstÃ©e Lauder           Fenty  Hegai and Ester   House of Tara
               3               12                6               11              42              40              10              11
            Iman             IPSA             Kate            Kuddy        L'OrÃ©al          LakmÃ©       LancÃ´me  Laws of Nature
               8                6                6                5              36               4              40              17
   Lotus Herbals              MAC Make Up For Ever      Maybelline            NARS           Nykaa          Olivia          Revlon
               4               42               40               54              13               5               4              22
             RMK         Shiseido       Shu Uemera  Trim & Prissy
               9                6               11               13

>
> print("Frequency Table using count():")
[1] "Frequency Table using count():"
> print(df %>% count(.data[[cat_var]]))
            brand  n
1       Addiction 17
2   Beauty Bakerie 30
3   Bharat & Doris  7
4      Black Opal 12
5        Black Up 18
6     Blue Heaven  2
7     Bobbi Brown 30
8        Colorbar  3
```

VIBHUTI_S079

```
Console   Terminal ×   Background Jobs ×
R 4.1.2 · ~/
>
> print("Frequency Table using count():")
[1] "Frequency Table using count():"
> print(df %>% count(.data[[cat_var]]))
              brand  n
1         Addiction 17
2    Beauty Bakerie 30
3    Bharat & Doris  7
4       Black Opal 12
5         Black Up 18
6       Blue Heaven  2
7       Bobbi Brown 30
8          Colorbar  3
9  Covergirl + Olay 12
10             Dior  6
11        Elsas Pro 11
12     EstÃ©e Lauder 42
13            Fenty 40
14  Hegai and Ester 10
15    House of Tara 11
16             IPSA  6
17             Iman  8
18             Kate  6
19            Kuddy  5
20          L'OrÃ©al 36
21           LakmÃ©  4
22          LancÃ´me 40
23   Laws of Nature 17
24    Lotus Herbals  4
25              MAC 42
26 Make Up For Ever 40
27       Maybelline 54
28             NARS 13
29            Nykaa  5
30           Olivia  4
31              RMK  9
32           Revlon 22
33         Shiseido  6
34       Shu Uemera 11
35    Trim & Prissy 13
36     bareMinerals 29
>
```

2) Generating frequency tables using table() or count() (R).

INPUT:-

library(psych)

df <- read.csv("shades.csv")

print("---- 1. Descriptive Statistics for Shades Dataset ----")

print("Summary of a numeric variable:")

summary(df[sapply(df, is.numeric)][,1])

print("Detailed Description of numeric variables:")

describe(df[sapply(df, is.numeric)])

OUTPUT:-

```
> library(psych)
>
> df <- read.csv("shades.csv")
>
> print("---- 1. Descriptive Statistics for Shades Dataset ----")
[1] "---- 1. Descriptive Statistics for Shades Dataset ----"
>
> print("Summary of a numeric variable:")
[1] "Summary of a numeric variable:"
> summary(df[sapply(df, is.numeric)][,1])
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   4.00   23.00   26.00   25.31   29.00   45.00      12
>
> print("Detailed Description of numeric variables:")
[1] "Detailed Description of numeric variables:"
> describe(df[sapply(df, is.numeric)])
      vars   n  mean    sd median trimmed   mad  min max range  skew kurtosis   se
H        1 613 25.31  5.33  26.00   25.58  4.45  4.0  45  41.0 -0.53     1.30 0.22
S        2 613  0.46  0.15   0.44    0.45  0.16  0.1   1   0.9  0.38    -0.17 0.01
V        3 613  0.78  0.17   0.84    0.80  0.13  0.2   1   0.8 -1.06     0.42 0.01
L        4 625 65.92 17.51  71.00   67.56 16.31 11.0  95  84.0 -0.80    -0.11 0.70
group    5 625  3.47  1.98   3.00    3.43  1.48  0.0   7   7.0  0.22    -0.88 0.08
>
```

3) Creating cross-tabulations and two-way tables using table() (R)

INPUT:-

df <- read.csv("shades.csv")

print("---- 3. Cross-Tabulation for Shades Dataset ----")

cat_vars <- names(df)[!sapply(df, is.numeric)]

cross_tab <- table(df[[cat_vars[1]]], df[[cat_vars[2]]])

print(cross_tab)

OUTPUT:-

VIBHUTI_S079

```
> df <- read.csv("shades.csv")
>
> print("---- 3. Cross-Tabulation for Shades Dataset ----")
[1] "---- 3. Cross-Tabulation for Shades Dataset ----"
>
> cat_vars <- names(df)[!sapply(df, is.numeric)]
>
> cross_tab <- table(df[[cat_vars[1]]], df[[cat_vars[2]]])
>
> print(cross_tab)

                   ad bb bd bh bm bo br bu cb di el ep fe he ht im ip ka ku la lc lh ln lo ma mb mu na ny oc ol rmk rv sh su tp
  Addiction        17  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0   0  0  0  0  0
  bareMinerals      0  0  0  0 29  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0   0  0  0  0  0
  Beauty Bakerie    0 30  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0   0  0  0  0  0
  Bharat & Doris    0  0  7  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0   0  0  0  0  0
  Black Opal        0  0  0  0  0 12  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0   0  0  0  0  0
  Black Up          0  0  0  0  0  0  0 18  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0   0  0  0  0  0
  Blue Heaven       0  0  0  2  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0   0  0  0  0  0
  Bobbi Brown       0  0  0  0  0  0 30  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0   0  0  0  0  0
  Colorbar          0  0  0  0  0  0  0  0  3  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0   0  0  0  0  0
  Covergirl + Olay  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0 12  0   0  0  0  0  0
  Dior              0  0  0  0  0  0  0  0  0  6  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0   0  0  0  0  0
  Elsas Pro         0  0  0  0  0  0  0  0  0  0  0 11  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0   0  0  0  0  0
  EstÃ©e Lauder     0  0  0  0  0  0  0  0  0  0 42  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0   0  0  0  0  0
  Fenty             0  0  0  0  0  0  0  0  0  0  0  0 40  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0   0  0  0  0  0
  Hegai and Ester   0  0  0  0  0  0  0  0  0  0  0  0  0 10  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0   0  0  0  0  0
  House of Tara     0  0  0  0  0  0  0  0  0  0  0  0  0  0 11  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0   0  0  0  0  0
  Iman              0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  8  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0   0  0  0  0  0
  IPSA              0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  6  0  0  0  0  0  0  0  0  0  0  0  0  0  0   0  0  0  0  0
  Kate              0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  6  0  0  0  0  0  0  0  0  0  0  0  0  0   0  0  0  0  0
  Kuddy             0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  5  0  0  0  0  0  0  0  0  0  0  0  0   0  0  0  0  0
  L'OrÃ©al          0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0 36  0  0  0  0  0  0  0   0  0  0  0  0
  LakmÃ©            0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  4  0  0  0  0  0  0  0  0  0  0  0   0  0  0  0  0
  LancÃ´me          0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0 40  0  0  0  0  0  0  0  0  0  0   0  0  0  0  0
  Laws of Nature    0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0 17  0  0  0  0  0  0  0  0   0  0  0  0  0
  Lotus Herbals     0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  4  0  0  0  0  0  0  0  0  0   0  0  0  0  0
  MAC               0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0 42  0  0  0  0  0  0   0  0  0  0  0
  Make Up For Ever  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0 40  0  0  0  0   0  0  0  0  0
 [ reached getOption("max.print") -- omitted 9 rows ]
>
```

4) Performing one-sample t-tests using t.test() (R).

INPUT:-

df <- read.csv("shades.csv")

print("---- 4. One-Sample t-test for Shades Dataset ----")

num_var <- names(df)[sapply(df, is.numeric)][1]

test_value <- 100

t_test_one <- t.test(df[[num_var]], mu = test_value)

print(t_test_one)

OUTPUT:-

VIBHUTI_S

```
> df <- read.csv("shades.csv")
>
> print("---- 4. One-Sample t-test for Shades Dataset ----")
[1] "---- 4. One-Sample t-test for Shades Dataset ----"
>
> num_var <- names(df)[sapply(df, is.numeric)][1]
>
> test_value <- 100
>
> t_test_one <- t.test(df[[num_var]], mu = test_value)
>
> print(t_test_one)

        One Sample t-test

data:  df[[num_var]]
t = -347.07, df = 612, p-value < 2.2e-16
alternative hypothesis: true mean is not equal to 100
95 percent confidence interval:
 24.89224 25.73745
sample estimates:
mean of x
 25.31485

> |
```

5) Performing independent two-sample t-tests using t.test() with grouping (R).

INPUT:-

```
library(dplyr)


shdes <- read.csv("shades.csv")


head(shdes)


print("---- 5. Independent Two-Sample t-test ----")


num_var <- "H"

binary_var <- "group"




if(length(unique(shdes[[binary_var]])) > 2){

  shdes[[binary_var]] <- ifelse(shdes[[binary_var]] > median(shdes[[binary_var]]), "High",
"Low")
```

VIBHUTI_

}

shdes[[binary_var]] <- as.factor(shdes[[binary_var]])

t_test_two <- t.test(shdes[[num_var]] ~ shdes[[binary_var]])

print(t_test_two)

OUTPUT:-

```
> library(dplyr)
>
> shdes <- read.csv("shades.csv")
>
> head(shdes)
      brand brand_short product product_short    hex  H    S    V  L group
1 Maybelline          mb  Fit Me           fmf f3cfb3 26 0.26 0.95 86     2
2 Maybelline          mb  Fit Me           fmf ffe3c2 32 0.24 1.00 92     2
3 Maybelline          mb  Fit Me           fmf ffe0cd 23 0.20 1.00 91     2
4 Maybelline          mb  Fit Me           fmf ffd3be 19 0.25 1.00 88     2
5 Maybelline          mb  Fit Me           fmf bd9584 18 0.30 0.74 65     2
6 Maybelline          mb  Fit Me           fmf eabda6 20 0.29 0.92 80     2
>
> print("---- 5. Independent Two-Sample t-test ----")
[1] "---- 5. Independent Two-Sample t-test ----"
>
> num_var <- "H"
> binary_var <- "group"
>
>
> if(length(unique(shdes[[binary_var]])) > 2){
+   shdes[[binary_var]] <- ifelse(shdes[[binary_var]] > median(shdes[[binary_var]]), "Hig
h", "Low")
+ }
> shdes[[binary_var]] <- as.factor(shdes[[binary_var]])
>
> t_test_two <- t.test(shdes[[num_var]] ~ shdes[[binary_var]])
>
> print(t_test_two)

        Welch Two Sample t-test

data:  shdes[[num_var]] by shdes[[binary_var]]
t = 3.3099, df = 610.62, p-value = 0.0009882
alternative hypothesis: true difference in means between group High and group Low is not eq
ual to 0
95 percent confidence interval:
```

```
R 4.1.2 · ~/
>
> shdes <- read.csv("shades.csv")
>
> head(shdes)
      brand brand_short product product_short    hex  H    S    V  L group
1 Maybelline          mb  Fit Me           fmf f3cfb3 26 0.26 0.95 86     2
2 Maybelline          mb  Fit Me           fmf ffe3c2 32 0.24 1.00 92     2
3 Maybelline          mb  Fit Me           fmf ffe0cd 23 0.20 1.00 91     2
4 Maybelline          mb  Fit Me           fmf ffd3be 19 0.25 1.00 88     2
5 Maybelline          mb  Fit Me           fmf bd9584 18 0.30 0.74 65     2
6 Maybelline          mb  Fit Me           fmf eabda6 20 0.29 0.92 80     2
>
> print("---- 5. Independent Two-Sample t-test ----")
[1] "---- 5. Independent Two-Sample t-test ----"
>
> num_var <- "H"
> binary_var <- "group"
>
>
> if(length(unique(shdes[[binary_var]])) > 2){
+   shdes[[binary_var]] <- ifelse(shdes[[binary_var]] > median(shdes[[binary_var]]), "Hig
h", "Low")
+ }
> shdes[[binary_var]] <- as.factor(shdes[[binary_var]])
>
> t_test_two <- t.test(shdes[[num_var]] ~ shdes[[binary_var]])
>
> print(t_test_two)

        Welch Two Sample t-test

data:  shdes[[num_var]] by shdes[[binary_var]]
t = 3.3099, df = 610.62, p-value = 0.0009882
alternative hypothesis: true difference in means between group High and group Low is not eq
ual to 0
95 percent confidence interval:
 0.5736462 2.2474929
sample estimates:
mean in group High  mean in group Low 
          26.05119           24.64062 

> |
```

```
Data
shades
shdes
t_test_
winequa
Values
binary_
num_var
```

6) Performing paired t-tests using t.test(paired=TRUE) (R).

INPUT:-

df <- read.csv("shades.csv")

print("---- 6. Paired t-test for Shades Dataset ----")

num_var <- names(df)[sapply(df, is.numeric)][1]

set.seed(123)

before_values <- df[[num_var]] - runif(nrow(df), min = 0.1, max = 0.5)

after_values <- df[[num_var]]

```
t_test_paired <- t.test(
  before_values,
  after_values,
  paired = TRUE
)

print(t_test_paired)
```

OUTPUT:-

```
> df <- read.csv("shades.csv")
>
> print("---- 6. Paired t-test for Shades Dataset ----")
[1] "---- 6. Paired t-test for Shades Dataset ----"
>
> num_var <- names(df)[sapply(df, is.numeric)][1]
>
> set.seed(123)
>
> before_values <- df[[num_var]] - runif(nrow(df), min = 0.1, max = 0.5)
> after_values  <- df[[num_var]]
>
> t_test_paired <- t.test(
+   before_values,
+   after_values,
+   paired = TRUE
+ )
>
> print(t_test_paired)

	Paired t-test

data:  before_values and after_values
t = -64.168, df = 612, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.3095536 -0.2911685
sample estimates:
mean of the differences
              -0.300361

> |
```

VIBHUTI