

RWorksheet_Salvador#3b

2023-10-17

```
#1.Create a data frame using the table below.
#a. Write the codes.
respo <- c(1:20)
sex <- c(2,2,1,2,2,2,2,2,2,1,2,2,2,2,2,2,1,2)
occu <- c(1,3,3,3,1,2,3,1,1,1,3,2,1,3,3,1,3,1,2,1)
pers_at_home <- c(5,7,3,8,5,9,6,7,8,4,7,5,4,7,8,8,3,11,7,6)
sibs <- c(6,4,4,1,2,1,5,3,1,2,3,2,5,5,2,1,2,5,3,2)
t_of_house <- c(1,2,3,1,1,3,3,1,2,3,2,3,2,2,3,3,3,3,3,2)
household_data <- data.frame(
  Respondents = respo,
  Sex = sex,
  FatherOccupation = occu,
  PersonAtHome = pers_at_home,
  SiblingsAtSchool = sibs,
  TypesofHouse = t_of_house
)
household_data
```

##	Respondents	Sex	FatherOccupation	PersonAtHome	SiblingsAtSchool	TypesofHouse
## 1	1	2	1	5	6	1
## 2	2	2	3	7	4	2
## 3	3	1	3	3	4	3
## 4	4	2	3	8	1	1
## 5	5	2	1	5	2	1
## 6	6	2	2	9	1	3
## 7	7	2	3	6	5	3
## 8	8	2	1	7	3	1
## 9	9	2	1	8	1	2
## 10	10	2	1	4	2	3
## 11	11	1	3	7	3	2
## 12	12	2	2	5	2	3
## 13	13	2	1	4	5	2
## 14	14	2	3	7	5	2
## 15	15	2	3	8	2	3
## 16	16	2	1	8	1	3
## 17	17	2	3	3	2	3
## 18	18	2	1	11	5	3
## 19	19	1	2	7	3	3
## 20	20	2	1	6	2	2

```
#1b. Describe the data. Get the structure or the summary of the data
str(household_data)
```

```
## 'data.frame': 20 obs. of 6 variables:
## $ Respondents : int 1 2 3 4 5 6 7 8 9 10 ...
## $ Sex : num 2 2 1 2 2 2 2 2 2 2 ...
```

```
## $ FatherOccupation: num 1 3 3 3 1 2 3 1 1 1 ...
## $ PersonAtHome : num 5 7 3 8 5 9 6 7 8 4 ...
## $ SiblingsAtSchool: num 6 4 4 1 2 1 5 3 1 2 ...
## $ TypesofHouse : num 1 2 3 1 1 3 3 1 2 3 ...
```

```
summary(household_data)
```

```
## Respondents Sex FatherOccupation PersonAtHome
## Min. : 1.00 Min. :1.00 Min. :1.00 Min. : 3.0
## 1st Qu.: 5.75 1st Qu.:2.00 1st Qu.:1.00 1st Qu.: 5.0
## Median :10.50 Median :2.00 Median :2.00 Median : 7.0
## Mean :10.50 Mean :1.85 Mean :1.95 Mean : 6.4
## 3rd Qu.:15.25 3rd Qu.:2.00 3rd Qu.:3.00 3rd Qu.: 8.0
## Max. :20.00 Max. :2.00 Max. :3.00 Max. :11.0
## SiblingsAtSchool TypesofHouse
## Min. :1.00 Min. :1.0
## 1st Qu.:2.00 1st Qu.:2.0
## Median :2.50 Median :2.5
## Mean :2.95 Mean :2.3
## 3rd Qu.:4.25 3rd Qu.:3.0
## Max. :6.00 Max. :3.0
```

#The data frame has 20 rows (observations) and 6 columns (variables).

#The variables are:

#Respondents - provides a unique numeric ID for each person surveyed.

#Sex - it represents the gender of the respondent, That 1 is for Male and 2 is for Female.

#Father's Occupation - represents the occupation of the fathers and the codes associated with each occupation.

#Persons at Home - it represents how many people are currently at home in each household.

#Siblings at School - which indicates how many siblings in the household are currently in school.

#Types of House - it represents different types of houses.

#1c. Is the mean number of siblings attending is 5?

```
sibs_mean <- mean(household_data$SiblingsAtSchool)
```

```
sibs_mean
```

```
## [1] 2.95
```

The mean of siblings attending is not 5, however its 2.95

#1d. Extract the 1st two rows and then all the columns using the subsetting functions. Write the codes and its result.

```
ExfirstTwoRows <- household_data[1:2,]
```

```
ExfirstTwoRows
```

```
## Respondents Sex FatherOccupation PersonAtHome SiblingsAtSchool TypesofHouse
## 1 1 2 1 5 6 1
## 2 2 2 3 7 4 2
```

#Output

```
#Respondents Sex FatherOccupation PersonAtHome SiblingsAtSchool TypesofHouse
```

```
#1 1 2 1 5 6 1
#2 2 2 3 7 4 2
```

#1e. Extract 3rd and 5th row with 2nd and 4th column. Write the codes and its result.

```
ExThirbandFifthRows <- household_data[c(3,5),c(2,4)]
```

```
ExThirbandFifthRows
```

```
## Sex PersonAtHome
## 3 1 3
```

```
## 5      2      5

#Output
# Sex PersonAtHome
#3      1      3
#5      2      5

#1f. Select the variable types of houses then store the vector that results as types_houses. Write the
types_houses <- household_data$TypesofHouse
types_houses

## [1] 1 2 3 1 1 3 3 1 2 3 2 3 2 2 3 3 3 3 2

#1g. Select only all Males respondent that their father occupation was farmer. Write the codes and its o
male_farmer <- household_data[household_data$Sex == 1 & household_data$FatherOccupation == 1,]
male_farmer

## [1] Respondents      Sex      FatherOccupation PersonAtHome
## [5] SiblingsAtSchool TypesofHouse
## <0 rows> (or 0-length row.names)

# 0 Obs.

#1h. Select only all females respondent that have greater than or equal to 5 number of siblings attendin
female_Respo <- household_data[household_data$SiblingsAtSchool >= 5,]
female_Respo

##      Respondents Sex FatherOccupation PersonAtHome SiblingsAtSchool TypesofHouse
## 1              1  2              1              5              6              1
## 7              7  2              3              6              5              3
## 13             13  2              1              4              5              2
## 14             14  2              3              7              5              2
## 18             18  2              1             11              5              3

# 5 obs
# Respondents Sex FatherOccupation PersonAtHome SiblingsAtSchool TypesofHouse
#1              1  2              1              5              6              1
#7              7  2              3              6              5              3
#13             13  2              1              4              5              2
#14             14  2              3              7              5              2
#18             18  2              1             11              5              3

#2. Write a R program to create an empty data frame. Using the following codes:
df = data.frame(Ints=integer(),
Doubles=double(), Characters=character(),
Logicals=logical(),
Factors=factor(),
stringsAsFactors=FALSE
)
print("Structure of the empty dataframe:")

## [1] "Structure of the empty dataframe:"
print(str(df))

## 'data.frame':    0 obs. of  5 variables:
## $ Ints      : int
## $ Doubles   : num
```

```
## $ Characters: chr
## $ Logicals : logi
## $ Factors : Factor w/ 0 levels:
## NULL

#df is a data frame that has been initialized with zero rows and five columns.
#The data types of the columns are as follows:
#ints are represented as integers.
#doubles are stored as double-precision floating-point numbers.
#characters contain character or text data.
#logicals are used for storing logical (True/False) values.
#factors are set as factors with zero levels, essentially making them empty.
#This empty data frame serves as a template and can be filled with data as needed.

#3.Create a .csv file of this. Save it as HouseholdData.csv
new_respo <- c(1:10)
new_sex <- c("Male", "Female", "Female", "Male", "Male", "Female", "Female", "Male", "Female", "Male")
new_occu <- c(1,2,3,3,1,2,2,3,1,3)
new_persathome <- c(5,7,3,8,6,4,4,2,11,6)
new_sibs <- c(2,3,0,5,2,3,1,2,6,2)
newt_of_house <- c("Wood", "Congrete", "Congrete", "Wood", "Semi-concrete", "Semi-concrete", "Wood", "S",
householdData <- data.frame(
  Respondents = new_respo,
  Sex = new_sex,
  FatherOccupation = new_occu,
  PersonAtHome = new_persathome,
  SiblingsAtSchool = new_sibs,
  TypesofHouse = newt_of_house
)
write.csv(householdData, file = "HouseholdData.csv")
#3a. Import the csv file into the R environment. Write the codes.
imported <- read.csv("HouseholdData.csv")
imported
```

```
##      X Respondents      Sex FatherOccupation PersonAtHome SiblingsAtSchool
## 1    1            1   Male                1             5             2
## 2    2            2 Female                2             7             3
## 3    3            3 Female                3             3             0
## 4    4            4   Male                3             8             5
## 5    5            5   Male                1             6             2
## 6    6            6 Female                2             4             3
## 7    7            7 Female                2             4             1
## 8    8            8   Male                3             2             2
## 9    9            9 Female                1            11             6
## 10  10           10   Male                3             6             2
##      TypesofHouse
## 1      Wood
## 2    Congrete
## 3    Congrete
## 4      Wood
## 5 Semi-concrete
## 6 Semi-concrete
## 7      Wood
## 8 Semi-concrete
## 9 Semi-concrete
```

```
## 10      Congrete
```

```
#3b. Convert the Sex into factor using factor() function and change it into integer. [Legend: Male = 1 and Female = 2]
imported$Sex <- factor(imported$Sex, levels = c("Male", "Female"))
imported$Sex <- as.integer(imported$Sex)
imported$Sex
```

```
## [1] 1 2 2 1 1 2 2 1 2 1
```

```
#3c. Convert the Type of Houses into factor and change it into integer. [Legend: Wood = 1; Congrete = 2; Semi-Congrete = 3]
imported$TypesofHouse <- factor(imported$TypesofHouse, levels = c("Wood", "Congrete", "Semi-Congrete"))
imported$TypesofHouse <- as.integer(imported$TypesofHouse)
imported$TypesofHouse
```

```
## [1] 1 2 2 1 NA NA 1 NA NA 2
```

```
#3d. On father's occupation, factor it as Farmer = 1; Driver = 2; and Others = 3. What is the R code and output?
imported$FatherOccupation <- factor(imported$FatherOccupation, levels = c(1,2,3), labels = c("Farmer", "Driver", "Others"))
imported$FatherOccupation
```

```
## [1] Farmer Driver Others Others Farmer Driver Driver Others Farmer Others
## Levels: Farmer Driver Others
```

```
#3e
female_driver <- imported[imported$Sex == 2 & imported$FatherOccupation == "Driver",]
female_driver
```

```
## X Respondents Sex FatherOccupation PersonAtHome SiblingsAtSchool TypesofHouse
## 2 2          2 2          Driver          7          3          2
## 6 6          6 2          Driver          4          3          NA
## 7 7          7 2          Driver          4          1          1
```

```
#3f
greaterFive <- imported[imported$SiblingsAtSchool >= 5,]
greaterFive
```

```
## X Respondents Sex FatherOccupation PersonAtHome SiblingsAtSchool TypesofHouse
## 4 4          4 1          Others          8          5          1
## 9 9          9 2          Farmer         11          6          NA
```

```
#4
```

```
# On July 14, there were more negative sentiments compared to the other sentiments. This could indicate a trend of negative sentiment over time.
# On July 15, all sentiments increased, with the negative sentiment as the highest. This could imply that the sentiment is becoming more positive over time.
# On July 17 and July 18, the negative sentiments stayed high and the neutral and positive sentiments rose.
# On July 20, all sentiments got to their lowest with but there were still more negative sentiments than the other sentiments.
# On July 21, experienced an increase in all sentiments, with the negative being the highest. This could indicate a trend of negative sentiment over time.
# From this data, we could assume that public sentiment is responsive to external factors and it also varies over time.
```