```r
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
#1. Create a data frame using the table given.
#a. Write the code
HouseholdData1<-data.frame(
Respondents=c(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20),
Sex=c(2,2,1,2,2,2,2,2,2,2,1,2,2,2,2,2,2,2,1,2),
Fathers_Occupation=c(1,3,3,3,1,2,3,1,1,1,3,2,1,3,3,1,3,1,2,1), Persons_At_Home=c(5,7,3,8,5,9,6,7,8,4,7,5
)
HouseholdData1
```

```
##    Respondents Sex Fathers_Occupation Persons_At_Home Siblings_At_School
## 1            1   2                  1               5                  6
## 2            2   2                  3               7                  4
## 3            3   1                  3               3                  4
## 4            4   2                  3               8                  1
## 5            5   2                  1               5                  2
## 6            6   2                  2               9                  1
## 7            7   2                  3               6                  5
## 8            8   2                  1               7                  3
## 9            9   2                  1               8                  1
## 10          10   2                  1               4                  2
## 11          11   1                  3               7                  3
## 12          12   2                  2               5                  2
## 13          13   2                  1               4                  5
## 14          14   2                  3               7                  5
## 15          15   2                  3               8                  2
## 16          16   2                  1               8                  1
## 17          17   2                  3               3                  2
## 18          18   2                  1              11                  5
## 19          19   1                  2               7                  3
## 20          20   2                  1               6                  2
##    Type_Of_Houses
## 1               1
## 2               2
## 3               3
## 4               1
## 5               1
## 6               3
## 7               3
## 8               1
```

```
## 9               2
## 10              3
## 11              2
## 12              3
## 13              2
## 14              2
## 15              3
## 16              3
## 17              3
## 18              3
## 19              3
## 20              2
```

```
#b. Describe the data. Get the structure or the summary of the data
summary(HouseholdData1)
```

```
##   Respondents          Sex       Fathers_Occupation Persons_At_Home
##  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
##  Siblings_At_School Type_Of_Houses
##  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
```

```
#c. Is the mean number of siblings attending is 5?
#Answer: No, the mean number is 2.95.
siblings_mean<-mean(HouseholdData1$Siblings_At_School)
siblings_mean
```

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

```
#d. Extract the 1st two rows and then all the columns using the subsetting functions.#Write the codes a
subset_HD<-subset(HouseholdData1[1:2,2:6])
subset_HD
```

```
##   Sex Fathers_Occupation Persons_At_Home Siblings_At_School Type_Of_Houses
## 1   2                  1               5                  6              1
## 2   2                  3               7                  4              2
```

```
#e. Extract 3rd and 5th row with 2nd and 4th column. Write the codes and its result.
subset_HD2<-subset(HouseholdData1[c(3,5), c(2,4)])
subset_HD2
```

```
##   Sex Persons_At_Home
## 3   1               3
## 5   2               5
```

```r
#f. Select the variable types of houses then store the vector that results as types_houses. Write the c
types_houses<-HouseholdData1[c(6)]
types_houses
```

```
##    Type_Of_Houses
## 1               1
## 2               2
## 3               3
## 4               1
## 5               1
## 6               3
## 7               3
## 8               1
## 9               2
## 10              3
## 11              2
## 12              3
## 13              2
## 14              2
## 15              3
## 16              3
## 17              3
## 18              3
## 19              3
## 20              2
```

```r
#g. Select only all Males respondent that their father occupation was farmer. Write the codes and its o
selected_data<-HouseholdData1%>% select(1,2,3)
maFarmer<-selected_data[HouseholdData1$Sex==1,]
maFarmer
```

```
##    Respondents Sex Fathers_Occupation
## 3            3   1                  3
## 11          11   1                  3
## 19          19   1                  2
```

```r
#h. Select only all females respondent that have greater than or equal to 5 number of siblings attendin
selected_data2<-HouseholdData1%>% select(1,2,5)
feSiblings<-selected_data2[HouseholdData1$Siblings_At_School>=5,]
feSiblings
```

```
##    Respondents Sex Siblings_At_School
## 1            1   2                  6
## 7            7   2                  5
## 13          13   2                  5
## 14          14   2                  5
## 18          18   2                  5
```

```r
colnames(HouseholdData1)<-c("Respondents", "Sex", "Fathers Occupation", "Persons at Home", "Siblings at
HouseholdData1
```

```
##    Respondents Sex Fathers Occupation Persons at Home Siblings at School
## 1            1   2                  1               5                  6
## 2            2   2                  3               7                  4
## 3            3   1                  3               3                  4
## 4            4   2                  3               8                  1
## 5            5   2                  1               5                  2
## 6            6   2                  2               9                  1
## 7            7   2                  3               6                  5
## 8            8   2                  1               7                  3
## 9            9   2                  1               8                  1
## 10          10   2                  1               4                  2
## 11          11   1                  3               7                  3
## 12          12   2                  2               5                  2
## 13          13   2                  1               4                  5
## 14          14   2                  3               7                  5
## 15          15   2                  3               8                  2
## 16          16   2                  1               8                  1
## 17          17   2                  3               3                  2
## 18          18   2                  1              11                  5
## 19          19   1                  2               7                  3
## 20          20   2                  1               6                  2
##    Types of Houses
## 1                1
## 2                2
## 3                3
## 4                1
## 5                1
## 6                3
## 7                3
## 8                1
## 9                2
## 10               3
## 11               2
## 12               3
## 13               2
## 14               2
## 15               3
## 16               3
## 17               3
## 18               3
## 19               3
## 20               2
```

```r
#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:"
```

```r
print(str(df))
```

```
## 'data.frame':    0 obs. of  5 variables:
##  $ Ints      : int
##  $ Doubles   : num
##  $ Characters: chr
##  $ Logicals  : logi
##  $ Factors   : Factor w/ 0 levels:
## NULL
```

```r
#a. Describe the results. ANSWER: It is a structure of an empty data frame. Since the data frame is emp

#3. Create a .csv file of this. Save it as HouseholdData.csv

#a. Import the csv file into the R environment. Write the codes.


RespondentsNew<-c(1,2,3,4,5,6,7,8,9,10)
SexNew<-c("Male", "Female", "Female", "Male", "Male", "Female", "Female", "Male", "Female", "Male")
FathersOccupationNew<-c(1,2,3,3,1,2,2,3,1,3)
PeAtHomeNew<-c(5,7,3,8,6,4,4,2,11,6)
SibAtSchoolNew<-c(2,3,0,5,2,3,1,2,6,2)
TypesofHousesNew<-c("Wood","Congrete","Congrete","Wood", "Semi-Congrete","Semi-Congrete","Wood","Semi-C
HouseholdData<-data.frame(
  RespondentsNew,
  SexNew,
  FathersOccupationNew,
  PeAtHomeNew,
  SibAtSchoolNew,
  TypesofHousesNew,
  stringsAsFactors=FALSE
  )
HouseholdData
```

```
##    RespondentsNew SexNew FathersOccupationNew PeAtHomeNew SibAtSchoolNew
## 1               1   Male                    1           5              2
## 2               2 Female                    2           7              3
## 3               3 Female                    3           3              0
## 4               4   Male                    3           8              5
## 5               5   Male                    1           6              2
## 6               6 Female                    2           4              3
## 7               7 Female                    2           4              1
## 8               8   Male                    3           2              2
## 9               9 Female                    1          11              6
## 10             10   Male                    3           6              2
##    TypesofHousesNew
## 1              Wood
## 2          Congrete
## 3          Congrete
## 4              Wood
## 5     Semi-Congrete
## 6     Semi-Congrete
## 7              Wood
```

```
## 8     Semi-Congrete
## 9     Semi-Congrete
## 10         Congrete
```

```
csv.file<-"HouseholdData.csv"
HouseholdData<-read.csv("HouseholdData.csv")
HouseholdData
```

```
##     X RespondentsNew SexNew FathersOccupationNew PeAtHomeNew SibAtSchoolNew
## 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
##    TypesofHousesNew
## 1              Wood
## 2          Congrete
## 3          Congrete
## 4              Wood
## 5     Semi-Congrete
## 6     Semi-Congrete
## 7              Wood
## 8     Semi-Congrete
## 9     Semi-Congrete
## 10         Congrete
```

```
#b. Convert the Sex into factor using factor() function and change it into integer.[Legend:
#Male = 1 and Female = 2]. Write the R codes and its output.
unique(HouseholdData$SexNew)
```

```
## [1] "Male"   "Female"
```

```
HD1 <- factor(HouseholdData$SexNew, levels = c("Male", "Female"))
legend_mapping1<-c("Male"=1, "Female"=2)
HD1<-as.integer(legend_mapping1[HouseholdData$SexNew])
unique(HD1)
```

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

```
HD1
```

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

```
#c. Convert the Type of Houses into factor and change it into integer. [Legend: Wood
#= 1; Congrete = 2; Semi-Congrete = 3]. Write the R codes and its output.
unique(HouseholdData$TypesofHousesNew)
```

```
## [1] "Wood"          "Congrete"      "Semi-Congrete"
```

```
HD2<-factor(HouseholdData$TypesofHousesNew, levels=c("Wood", "Congrete", "Semi-Congrete"))
legend_mapping2<-c("Wood"=1, "Congrete"=2, "Semi-Congrete"=3)
HD2<-as.integer(legend_mapping2[HouseholdData$TypesofHousesNew])
unique(HD2)
```

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

```
HD2
```

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

```
#d. On father's occupation, factor it as Farmer = 1; Driver = 2; and Others = 3. What is the R code and
unique(HouseholdData$FathersOccupationNew)
```

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

```
HD3<-factor(HouseholdData$FathersOccupationNew, levels=c("Farmer", "Driver", "Others"))
legend_mapping3<-c("Farmer"=1, "Driver"=2, "Others"=3)
HD3<-as.integer(legend_mapping3[HouseholdData$FathersOccupationNew])
unique(HD3)
```

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

```
HD3
```

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

```
#e. Select only all females respondent that has a father whose occupation is driver. Write the codes an
selectedHD<-HouseholdData%>% select(1,2,3)
feFather<-selectedHD[HouseholdData$FathersOccupationNew==2,]
feFather
```

```
##   X RespondentsNew SexNew
## 2 2              2 Female
## 6 6              6 Female
## 7 7              7 Female
```

```
#f. Select the respondents that have greater than or equal to 5 number of siblings attending school. Wr
selectedHD2<-HouseholdData%>%select(1,5)
ReSiblings<-selectedHD2[HouseholdData$SibAtSchoolNew>=5,]
ReSiblings
```

```
##   X PeAtHomeNew
## 4 4           8
## 9 9          11
```

```r
colnames(HouseholdData)<-c("Respondents", "Sex", "Fathers Occupation", "Persons at Home", "Siblings at S
HouseholdData
```

```
##    Respondents Sex Fathers Occupation Persons at Home Siblings at School
## 1            1   1               Male              1                  5
## 2            2   2             Female              2                  7
## 3            3   3             Female              3                  3
## 4            4   4               Male              3                  8
## 5            5   5               Male              1                  6
## 6            6   6             Female              2                  4
## 7            7   7             Female              2                  4
## 8            8   8               Male              3                  2
## 9            9   9             Female              1                 11
## 10          10  10               Male              3                  6
##    Types of Houses            NA
## 1                2          Wood
## 2                3      Congrete
## 3                0      Congrete
## 4                5          Wood
## 5                2 Semi-Congrete
## 6                3 Semi-Congrete
## 7                1          Wood
## 8                2 Semi-Congrete
## 9                6 Semi-Congrete
## 10               2      Congrete
```

```r
#4. Interpret the graph.

#The graph, titled "Sentiments of Tweet Per day" is a bar graph starting from
#the day of July 14 to July 20 of the year 2020.
#It is categorized by 3 sentiments namely:
#Negative(Red), Neutral(Yellow), and Positive(Blue) in which could reveal the mood or tone of the said

#Negative Sentiment: These tweets convey discontent, disapproval, or negative sentiments.
#This is the most noticeable sentiment being conveyed in the following days.
#On specific dates like July 15 and July 21, 2020, there was a notable uptick in negative tweets,
#indicating fervent discussions or concerns.

#Neutral Sentiment: These tweets maintain an even-handed and unbiased tone,
#presenting information objectively.
#Throughout July 2020, including days like July 14, 15, 17, 18, and 21,
#neutral sentiments were prominent, reflecting a range of non-partisan discussions.

#Positive Sentiment: This category encompasses tweets that radiate
#positivity, enthusiasm, and a hopeful perspective.
#Despite the presence of negative sentiments on specific days, such as July 14, 15, 17, 18, and 21,
#positive tweets also shone through, signifying resilience, hope,
#or an optimistic outlook amidst diverse sentiments.

#This graph effectively encapsulates Twitter's sentiment dynamics in July 2020.
#It underscores fluctuations in Negative, Neutral, and Positive sentiments on specific dates,
#offering a concise overview of the emotional landscape during that period.
```