Name: A Sai Charan
Reg_Number: 20BDS0354

SLOT:L3+L4
FACULTY: SUNILKUMAR

CSE-3046 Lab-Assignment-1 Programming For Data Science

Importing Libraries and Dataset

```
library(tidyverse)
```

```
## ── Attaching core tidyverse packages ──────────────────────── tidyverse
2.0.0 ──
## ✓ dplyr     1.1.2     ✓ readr     2.1.4
## ✓ forcats   1.0.0     ✓ stringr   1.5.0
## ✓ ggplot2   3.4.3     ✓ tibble    3.2.1
## ✓ lubridate 1.9.2     ✓ tidyr     1.3.0
## ✓ purrr     1.0.2
## ── Conflicts ────────────────────────────────────────
tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all
conflicts to become errors
```

```
AutoMobiles<-read.csv("C:/Users/saic3/CSE3046-F2-
LAB_SLOT_L3+L4/Datasets/Automobile.csv")
```

Check For Null Values:

```
ToNull<-is.na(AutoMobiles$price)
sum(ToNull)
```

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

Removing the NA Values

```
AutoMobiles<-AutoMobiles%>%fill(price,.direction = "down")
sum(is.na(AutoMobiles$price))
```

```
## [1] 0
```

Question-1:Find last n rows with price > 14000?

```
tail(AutoMobiles["price">14000])
```

```
##    index    company body.style wheel.base length engine.type
num.of.cylinders
## 56    80 volkswagen      sedan       97.3  171.7         ohc
four
## 57    81 volkswagen      sedan       97.3  171.7         ohc
```

```
four
## 58    82 volkswagen      sedan       97.3  171.7         ohc
four
## 59    86 volkswagen      sedan       97.3  171.7         ohc
four
## 60    87      volvo      sedan      104.3  188.8         ohc
four
## 61    88      volvo      wagon      104.3  188.8         ohc
four
##    horsepower average.mileage price
## 56         52              37  7775
## 57         85              27  7975
## 58         52              37  7995
## 59        100              26  9995
## 60        114              23 12940
## 61        114              23 13415
```

Question-2: Count the total company in the dataset?

```
unique(AutoMobiles$company)%>%length()

## [1] 16

unique(AutoMobiles$company)

##  [1] "alfa-romero"   "audi"          "bmw"           "chevrolet"
##  [5] "dodge"         "honda"         "isuzu"         "jaguar"
##  [9] "mazda"         "mercedes-benz" "mitsubishi"    "nissan"
## [13] "porsche"       "toyota"        "volkswagen"    "volvo"
```

Question-3: Find all the rows with body-style sedan?

```
AutoMobiles%>%filter(body.style=="sedan")

##    index       company body.style wheel.base length engine.type
## 1      3          audi      sedan       99.8  176.6         ohc
## 2      4          audi      sedan       99.4  176.6         ohc
## 3      5          audi      sedan       99.8  177.3         ohc
## 4      9           bmw      sedan      101.2  176.8         ohc
## 5     10           bmw      sedan      101.2  176.8         ohc
## 6     11           bmw      sedan      101.2  176.8         ohc
## 7     13           bmw      sedan      103.5  189.0         ohc
## 8     14           bmw      sedan      103.5  193.8         ohc
## 9     15           bmw      sedan      110.0  197.0         ohc
## 10    18     chevrolet      sedan       94.5  158.8         ohc
## 11    28         honda      sedan       96.5  175.4         ohc
## 12    29         honda      sedan       96.5  169.1         ohc
## 13    30         isuzu      sedan       94.3  170.7         ohc
## 14    31         isuzu      sedan       94.5  155.9         ohc
## 15    32         isuzu      sedan       94.5  155.9         ohc
```

```
## 16    33        jaguar      sedan      113.0  199.6        dohc
## 17    34        jaguar      sedan      113.0  199.6        dohc
## 18    35        jaguar      sedan      102.0  191.7        ohcv
## 19    43         mazda      sedan      104.9  175.0         ohc
## 20    44 mercedes-benz      sedan      110.0  190.9         ohc
## 21    46 mercedes-benz      sedan      120.9  208.1        ohcv
## 22    51    mitsubishi      sedan       96.3  172.4         ohc
## 23    52    mitsubishi      sedan       96.3  172.4         ohc
## 24    53        nissan      sedan       94.5  165.3         ohc
## 25    54        nissan      sedan       94.5  165.3         ohc
## 26    55        nissan      sedan       94.5  165.3         ohc
## 27    57        nissan      sedan      100.4  184.6        ohcv
## 28    80    volkswagen      sedan       97.3  171.7         ohc
## 29    81    volkswagen      sedan       97.3  171.7         ohc
## 30    82    volkswagen      sedan       97.3  171.7         ohc
## 31    86    volkswagen      sedan       97.3  171.7         ohc
## 32    87         volvo      sedan      104.3  188.8         ohc
##    num.of.cylinders horsepower average.mileage price
## 1              four        102              24 13950
## 2              five        115              18 17450
## 3              five        110              19 15250
## 4              four        101              23 16430
## 5              four        101              23 16925
## 6               six        121              21 20970
## 7               six        182              16 30760
## 8               six        182              16 41315
## 9               six        182              15 36880
## 10             four         70              38  6575
## 11             four        101              24 12945
## 12             four        100              25 10345
## 13             four         78              24  6785
## 14             four         70              38  6785
## 15             four         70              38  6785
## 16              six        176              15 32250
## 17              six        176              15 35550
## 18           twelve        262              13 36000
## 19             four         72              31 18344
## 20             five        123              22 25552
## 21            eight        184              14 40960
## 22             four         88              25  6989
## 23             four         88              25  8189
## 24             four         55              45  7099
## 25             four         69              31  6649
## 26             four         69              31  6849
## 27              six        152              19 13499
## 28             four         52              37  7775
## 29             four         85              27  7975
## 30             four         52              37  7995
## 31             four        100              26  9995
## 32             four        114              23 12940
```

```
AutoMobiles%>%filter(body.style=="sedan")%>%count()
```

```
##    n
## 1 32
```

Question-4: Find the 3rd most expensive car price and company name.

```
Q4<-
AutoMobiles%>%group_by(company)%>%summarise(newPrice=max(price))%>%arrange(desc(newPrice))
Q4
```

```
## # A tibble: 16 × 2
##    company       newPrice
##    <chr>            <int>
##  1 mercedes-benz    45400
##  2 bmw              41315
##  3 porsche          37028
##  4 jaguar           36000
##  5 audi             18920
##  6 mazda            18344
##  7 alfa-romero      16500
##  8 toyota           15750
##  9 nissan           13499
## 10 volvo            13415
## 11 honda            12945
## 12 volkswagen        9995
## 13 mitsubishi        8189
## 14 isuzu             6785
## 15 chevrolet         6575
## 16 dodge             6377
```

```
#Since Due to 0-based Indexing and No of Companies are 16 from the Q2 We Use 16-2 as Index
Q4[3,]
```

```
## # A tibble: 1 × 2
##   company newPrice
##   <chr>      <int>
## 1 porsche    37028
```

Question-5: Find the most expensive car for each company. #Similar to Question 4 Where Finding the Third Highest Car Price and Its Company

```
Q4
```

```
## # A tibble: 16 × 2
##    company       newPrice
##    <chr>            <int>
##  1 mercedes-benz    45400
##  2 bmw              41315
##  3 porsche          37028
##  4 jaguar           36000
##  5 audi             18920
##  6 mazda            18344
##  7 alfa-romero      16500
##  8 toyota           15750
##  9 nissan           13499
## 10 volvo            13415
## 11 honda            12945
## 12 volkswagen        9995
## 13 mitsubishi        8189
## 14 isuzu             6785
## 15 chevrolet         6575
## 16 dodge             6377
```

Question 6: Print all Toyota cars details

```
AutoMobiles%>%filter(company=="toyota")
```

```
##   index company body.style wheel.base length engine.type num.of.cylinders
## 1    66  toyota  hatchback       95.7  158.7         ohc             four
## 2    67  toyota  hatchback       95.7  158.7         ohc             four
## 3    68  toyota  hatchback       95.7  158.7         ohc             four
## 4    69  toyota      wagon       95.7  169.7         ohc             four
## 5    70  toyota      wagon       95.7  169.7         ohc             four
## 6    71  toyota      wagon       95.7  169.7         ohc             four
## 7    79  toyota      wagon      104.5  187.8        dohc              six
##   horsepower average.mileage price
## 1         62              35  5348
## 2         62              31  6338
## 3         62              31  6488
## 4         62              31  6918
## 5         62              27  7898
## 6         62              27  8778
## 7        156              19 15750
```

Question-7:Find the count of “convertible” type cars in “alfa-romero” company

```
count(AutoMobiles%>%filter(company=="alfa-romero"&body.style=="convertible"))
```

```
##   n
## 1 2
```

```
AutoMobiles%>%filter(company=="alfa-romero"&body.style=="convertible")
```

```
##   index     company  body.style wheel.base length engine.type
num.of.cylinders
## 1     0 alfa-romero convertible       88.6  168.8        dohc
four
## 2     1 alfa-romero convertible       88.6  168.8        dohc
four
##   horsepower average.mileage price
## 1        111              21 13495
## 2        111              21 16500
```

Question-8: Create a vector with 20 numeric items and extract top 2 most frequent items of a vector?

```
numeric_vector<-c(1, 2, 3, 2, 1, 4, 5, 1, 2, 6, 7, 7, 8, 9, 3, 10, 5, 5, 2,
1)
item_freq <- table(numeric_vector)
# Sort the frequencies in decreasing order
sorted_freq <- sort(item_freq, decreasing = TRUE)
# Extract the top 2 most frequent items
top_2_items <- as.numeric(names(sorted_freq[1:2]))

print(top_2_items)
```

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

Question-9:Create two dataframe with different attributes and merge them column wise.

```
df1<-
data.frame(Students=c("Somu","Venu","Venkat","Sri","Charvi","Marky","Duplex")
,Marks=c(10,100,49,40,15,21,95))
df1
```

```
##   Students Marks
## 1     Somu    10
## 2     Venu   100
## 3   Venkat    49
## 4      Sri    40
## 5   Charvi    15
## 6    Marky    21
## 7   Duplex    95
```

```
df2<-
data.frame(Items=c("Rice","Cofee","Tea","Oil","Vegetables","Fruits","Ghee"),P
rices=c(12,80,29,87,2,13,16))
df2
```

```
##        Items Prices
## 1       Rice     12
## 2      Cofee     80
## 3        Tea     29
## 4        Oil     87
## 5 Vegetables      2
## 6     Fruits     13
## 7       Ghee     16
```

```
df1<-cbind(df1,df2)
df1
```

```
##   Students Marks      Items Prices
## 1     Somu    10       Rice     12
## 2     Venu   100      Cofee     80
## 3   Venkat    49        Tea     29
## 4      Sri    40        Oil     87
## 5   Charvi    15 Vegetables      2
## 6    Marky    21     Fruits     13
## 7   Duplex    95       Ghee     16
```

Question-10: Create two dataframe with the same attributes and merge them row wise.

```
df1<-
data.frame(Students=c("Somu","Venu","Venkat","Sri","Charvi","Marky","Duplex")
,Marks=c(10,100,49,40,15,21,95))
df3<-data.frame(Students=c("Sai","Kent","Bruce"),Marks=c(1000,1,99))
df1<-rbind(df1,df3)
df1
```

```
##    Students Marks
## 1      Somu    10
## 2      Venu   100
## 3    Venkat    49
## 4       Sri    40
## 5    Charvi    15
## 6     Marky    21
## 7    Duplex    95
## 8       Sai  1000
## 9      Kent     1
## 10    Bruce    99
```

RMarkdownFile Link:https://drive.google.com/file/d/1cnEhewIg_wqU1vZ4S7zALaqI2EiD9KOQ/view?usp=sharing

HTML File Link:
https://drive.google.com/file/d/1vKRTWrZ4k4pOTnAZvU4Ou__XgFwQ4Qq-/view?usp=sharing