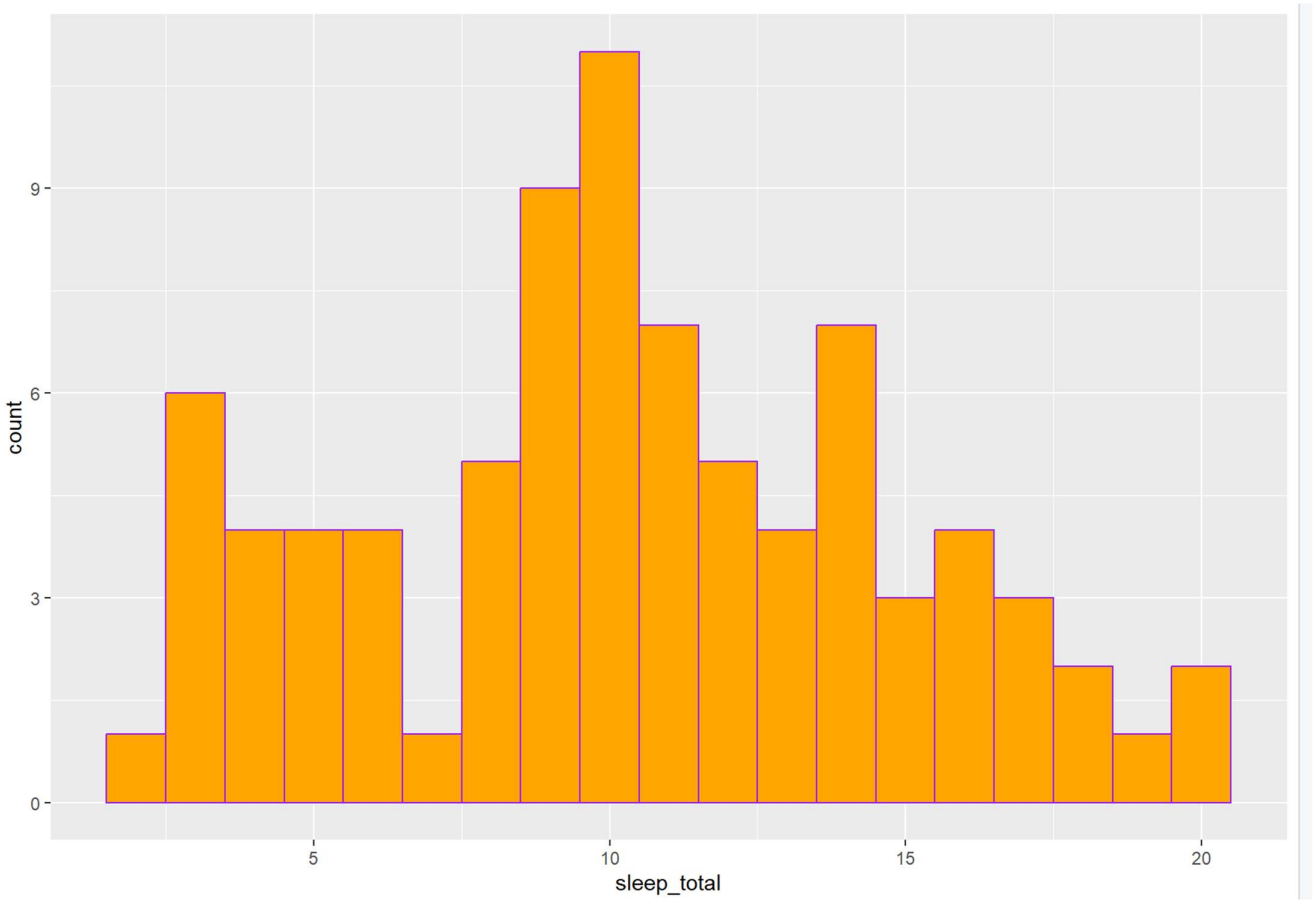
R Assignment 07(TA-2)

Name: Vidyut Chakrabarti Roll no. 68/CSE-B

Code:

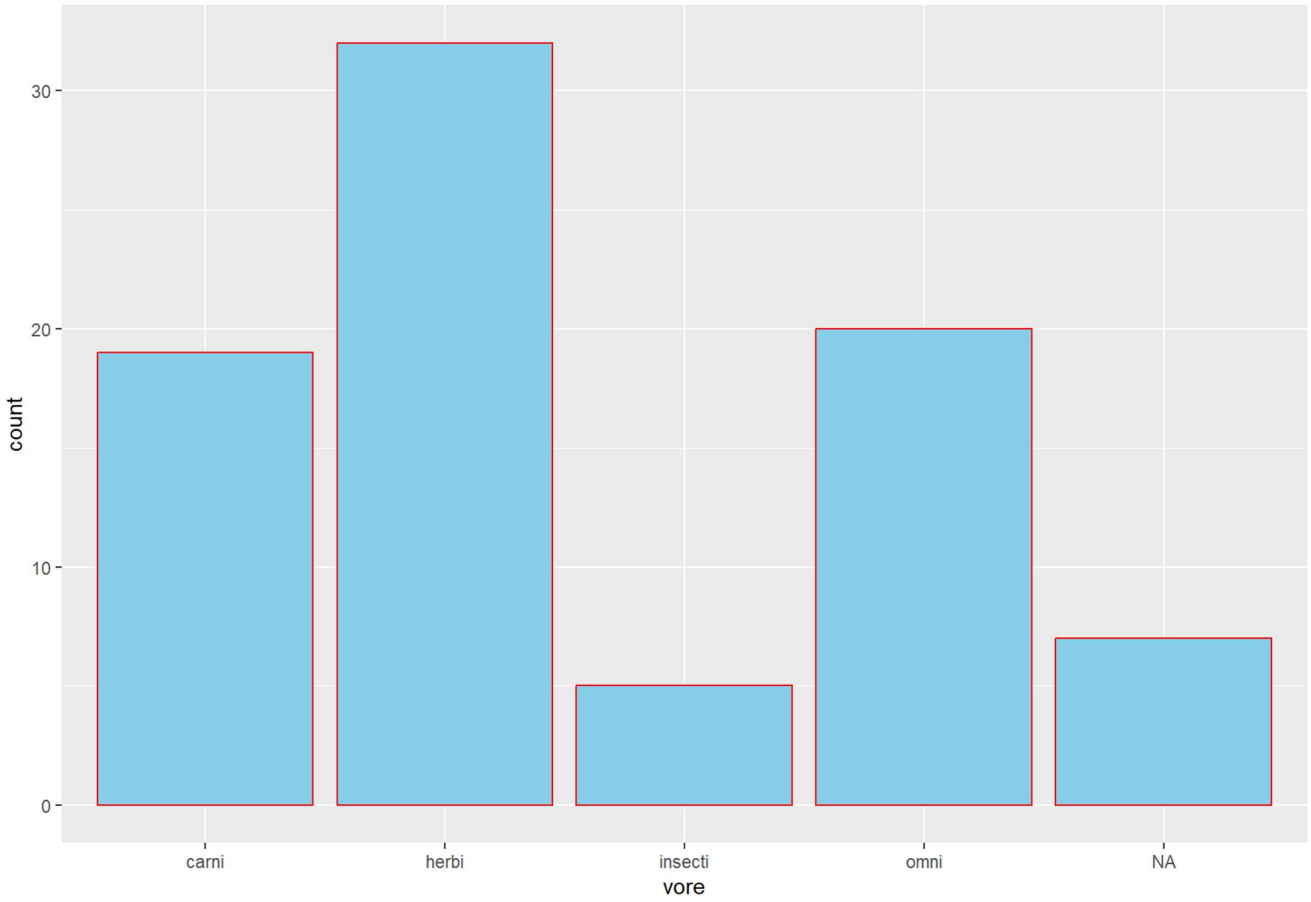
```
1. library(ggplot2)
2. > data("msleep")
3. > #histogram
4. > ggplot(msleep,aes(sleep_total)) +
5. +   geom_histogram(binwidth = 1,fill="orange",color="purple")
```

Output:




```
#bar_chart
ggplot(msleep,aes(vore)) +
  geom_bar(fill="skyblue",color="red")
```

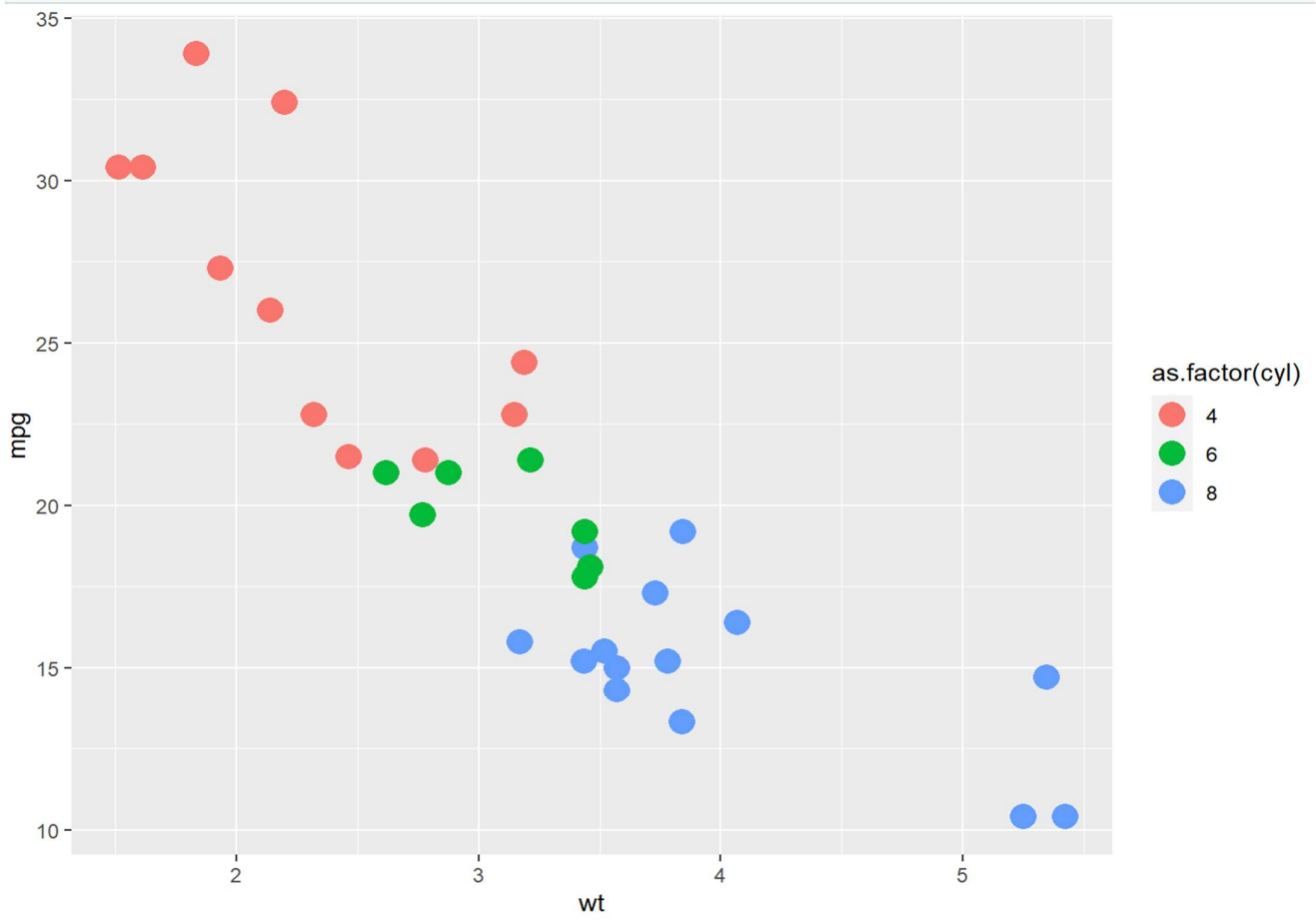
output:



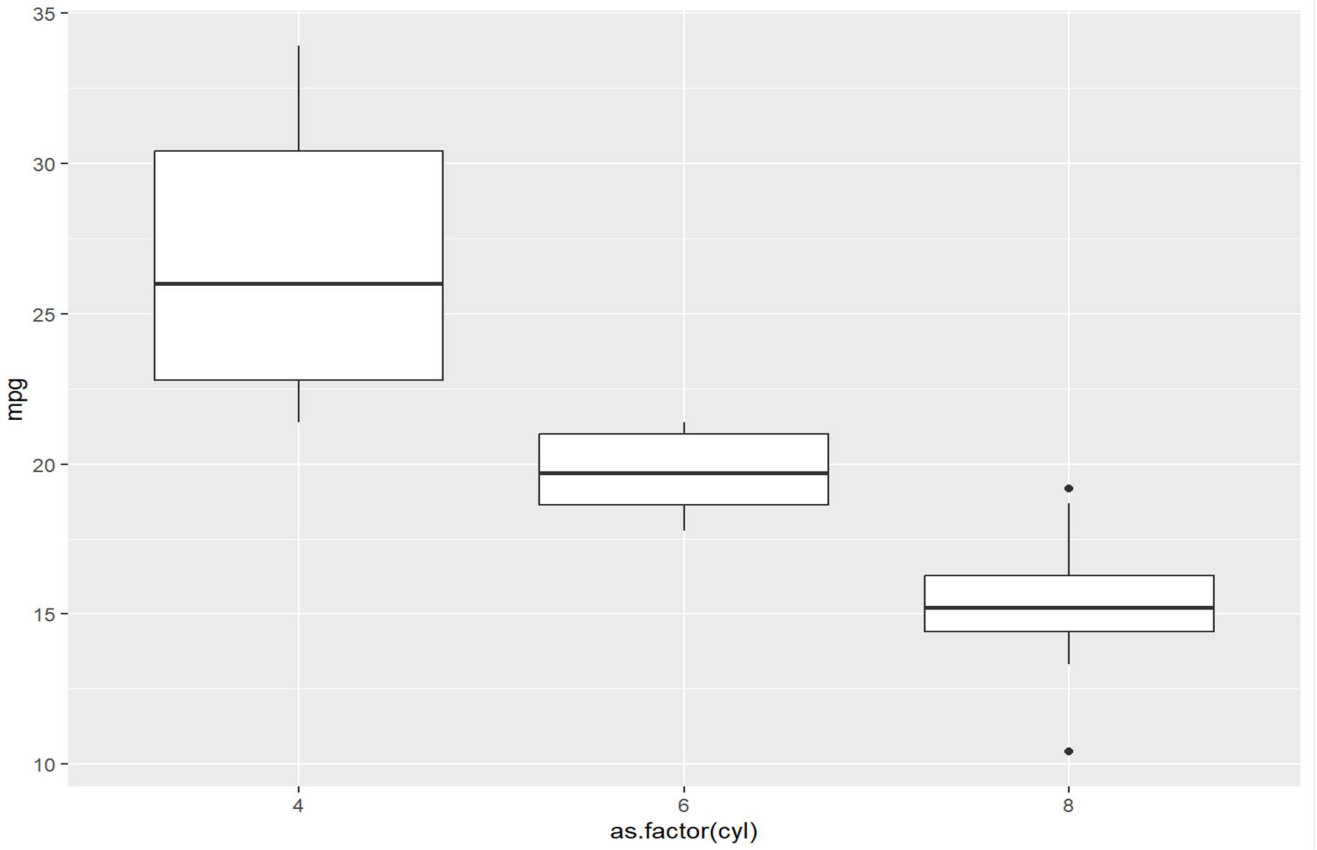
#structure
str(msleep)
output:

```
> str(msleep)
tibble [83 × 11] (S3: tbl_df/tbl/data.frame)
 $ name        : chr [1:83] "Cheetah" "Owl monkey" "Mountain beaver" "Greater short-tailed shrew" ..
 $ genus       : chr [1:83] "Acinonyx" "Aotus" "Aplodontia" "Blarina" ...
 $ vore        : chr [1:83] "carni" "omni" "herbi" "omni" ...
 $ order       : chr [1:83] "Carnivora" "Primates" "Rodentia" "Soricomorpha" ...
 $ conservation: chr [1:83] "lc" NA "nt" "lc" ...
 $ sleep_total : num [1:83] 12.1 17 14.4 14.9 4 14.4 8.7 7 10.1 3 ...
 $ sleep_rem   : num [1:83] NA 1.8 2.4 2.3 0.7 2.2 1.4 NA 2.9 NA ...
 $ sleep_cycle : num [1:83] NA NA NA 0.133 0.667 ...
 $ awake       : num [1:83] 11.9 7 9.6 9.1 20 9.6 15.3 17 13.9 21 ...
 $ brainwt     : num [1:83] NA 0.0155 NA 0.00029 0.423 NA NA NA 0.07 0.0982 ...
 $ bodywt      : num [1:83] 50 0.48 1.35 0.019 600 ...
```

data("mtcars")
> #scatter plots
> ggplot(mtcars,aes(wt,mpg,col=as.factor(cyl))) +
+ geom_point(size=5)
OUTPUT



#boxplot
> ggplot(mtcars, aes(as.factor(cyl),mpg)) +
+ geom_boxplot() OUTPUT:



#trying splitting and row binding:

```
> data("iris")
> iris_species<-split(iris,iris$Species)
> iris_setosa<-iris_species[[1]]
> iris_versicolor<-iris_species[[2]]
> iris_virginica<-iris_species[[3]]
```

OUTPUT: (Global environment window)

iris	150 obs. of 5 variables
iris_setosa	50 obs. of 5 variables
iris_species	List of 3
iris_versicolor	50 obs. of 5 variables
iris_virginica	50 obs. of 5 variables

```
head(iris_setosa)
head(iris_versicolor)
head(iris_virginica)
```

OUTPUT:

```
> head(iris_setosa)
  Sepal.Length Sepal.Width Petal.Length Petal.Width Species
1          5.1         3.5          1.4         0.2  setosa
2          4.9         3.0          1.4         0.2  setosa
3          4.7         3.2          1.3         0.2  setosa
4          4.6         3.1          1.5         0.2  setosa
5          5.0         3.6          1.4         0.2  setosa
6          5.4         3.9          1.7         0.4  setosa
> head(iris_versicolor)
   Sepal.Length Sepal.Width Petal.Length Petal.Width    Species
51          7.0         3.2          4.7         1.4 versicolor
52          6.4         3.2          4.5         1.5 versicolor
53          6.9         3.1          4.9         1.5 versicolor
54          5.5         2.3          4.0         1.3 versicolor
55          6.5         2.8          4.6         1.5 versicolor
56          5.7         2.8          4.5         1.3 versicolor
> head(iris_virginica)
    Sepal.Length Sepal.Width Petal.Length Petal.Width   Species
101          6.3         3.3          6.0         2.5 virginica
102          5.8         2.7          5.1         1.9 virginica
103          7.1         3.0          5.9         2.1 virginica
104          6.3         2.9          5.6         1.8 virginica
105          6.5         3.0          5.8         2.2 virginica
106          7.6         3.0          6.6         2.1 virginica
>
```

Importing dplyr to check and rbind.

```
species_back<-rbind.data.frame(iris_setosa,iris_versicolor,iris_virginica)
> library(dplyr)
> all.equal(iris,species_back)
```

OUTPUT: (global environment)

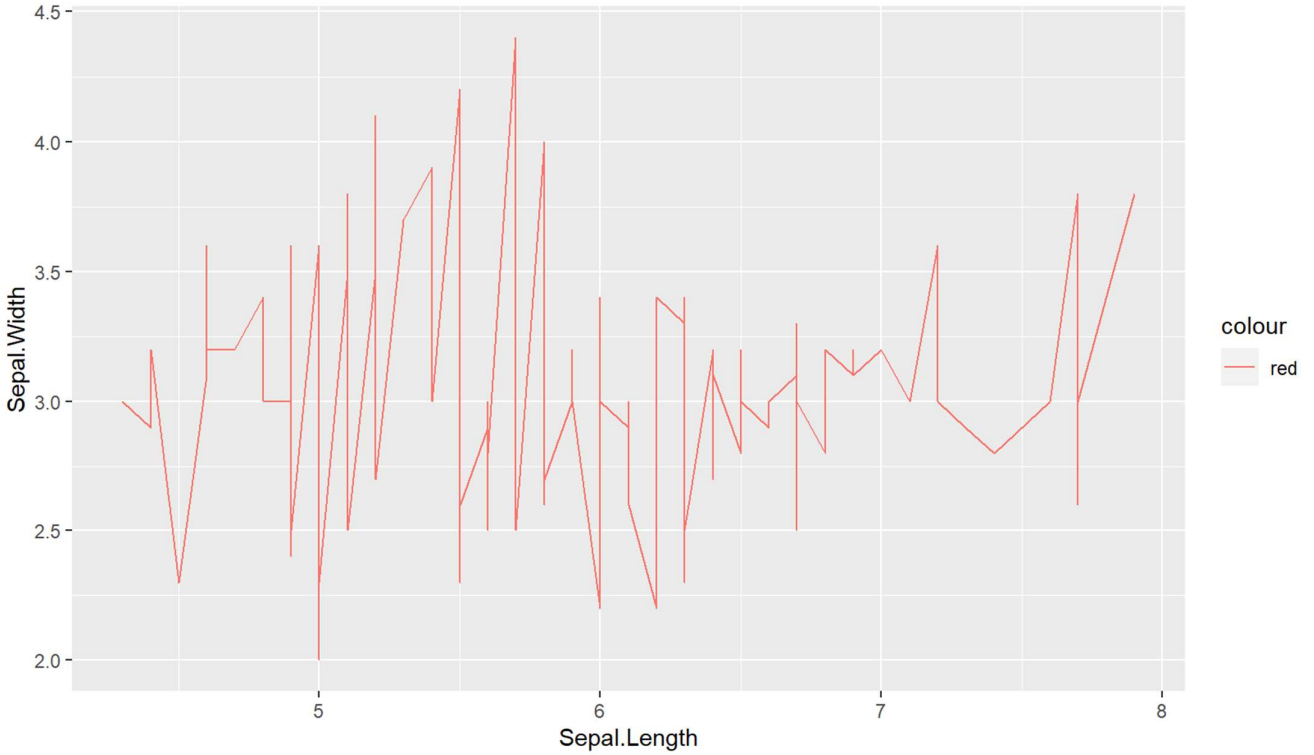
```
> all.equal(iris,species_back)
[1] TRUE
```

species_back	150 obs. of 5 variables

#line plot

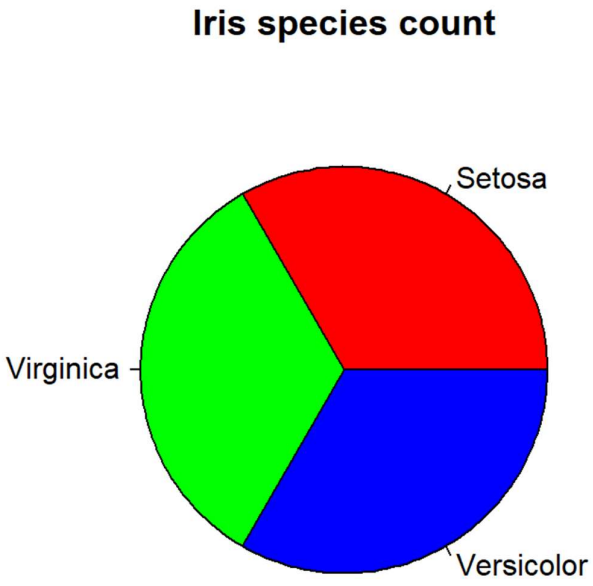
```
> ggplot(iris, aes(Sepal.Length,Sepal.Width,color="red")) +
+   geom_line()
```

OUTPUT:



#pie plot

```
> x <- c(50,50,50)
> labels <- c("Setosa","Virginica","Versicolor")
> pie(x, labels, main = "Iris species count", col = rainbow(length(x)))
```



Values	
labels	chr [1:3] "Setosa" "Virginica" "Versicolor"
x	num [1:3] 50 50 50

#importing csv
> std1<-read.csv("https://raw.githubusercontent.com/ShapeLab/ZooidsCompositePhysicalizations/master/Zooid_Vis/bin/data/student-dataset.csv")
> str(std1)
OUTPUT:

```
'data.frame':   307 obs. of  16 variables:
 $ id               : int  0 1 2 3 4 5 6 7 8 9 ...
 $ name             : chr  "Kiana Lor" "Joshua Lonaker" "Dakota Blanco" "Natasha Yarusso" ...
 $ nationality      : chr  "China" "United States of America" "United States of America" "United States of America"
 $ city             : chr  "Suzhou" "Santa Clarita" "Oakland" "Castro Valley" ...
 $ latitude         : num  31.3 34.4 37.8 37.7 -23.2 ...
 $ longitude        : num  120.6 -118.5 -122.3 -122.1 -45.9 ...
 $ gender           : chr  "F" "M" "F" "F" ...
 $ ethnic.group     : logi  NA NA NA NA NA NA ...
 $ age              : int  22 22 22 20 21 21 22 22 24 22 ...
 $ english.grade    : num  3.5 2.9 3.9 3.3 3.7 3.4 3.7 3.8 3.9 2.4 ...
 $ math.grade       : num  3.7 3.2 3.8 2.8 2.6 3.1 3.9 3.7 3.6 2.8 ...
 $ sciences.grade   : num  3.1 3.6 3.2 3.2 3.4 3.7 3.6 3.6 3.2 3.8 ...
 $ language.grade   : num  1 5 5 5 1 5 2 2 2 3 ...
 $ portfolio.rating : int  4 5 3 5 4 2 5 5 4 5 ...
 $ coverletter.rating: num  4 4 3 2 4 4 5 5 3 5 ...
 $ refletter.rating : int  4 5 4 4 5 4 5 4 5 4 ...
```

#deleting a column.
std1 <- subset(std1, select = -c(ethnic.group))

#using tapply to find mean scores of every nationality.
print("mean math scores for students from different countries:")
tapply(std1$math.grade, std1$nationality, mean)
print("mean sciences scores for students from different countries:")
tapply(std1$sciences.grade, std1$nationality, mean)
print("mean language scores for students from different countries:")
tapply(std1$language.grade, std1$nationality, mean)
print("mean english scores for students from different countries:")
tapply(std1$english.grade, std1$nationality, mean)

OUTPUT:

```
> print("mean sciences scores for students from different countries:")
[1] "mean sciences scores for students from different countries:"
> tapply(std1$sciences.grade, std1$nationality, mean)
             Bangladesh                  Brazil                  Canada                   Chile                   China 
               3.900000                3.055556                3.600000                4.000000                3.376923 
               Colombia                    Cuba       Dominican Republic                   Egypt             El Salvador 
               3.280000                3.200000                4.000000                3.200000                3.500000 
                Germany                   India                   Japan      Korea (Republic of)                  Mexico 
               3.900000                3.637500                3.492308                3.200000                3.441667 
                Morocco                 Myanmar             Netherlands               Nicaragua                Pakistan 
               3.800000                3.700000                3.700000                3.800000                3.333333 
                   Peru             Philippines                  Poland      Russian Federation                   Spain 
               3.000000                3.200000                3.000000                3.133333                3.400000 
               Thailand                 Tunisia                  Turkey                 Ukraine          United Kingdom 
               3.900000                4.000000                3.850000                3.900000                3.200000 
United States of America 
               3.455959 
```

OUTPUT:

```
> print("mean language scores for students from different countries:")
[1] "mean language scores for students from different countries:"
> tapply(std1$language.grade, std1$nationality, mean)
              Bangladesh                  Brazil                  Canada                   Chile                   China
                5.000000                2.888889                5.000000                3.000000                3.153846
                Colombia                    Cuba      Dominican Republic                   Egypt             El Salvador
                3.200000                3.000000                4.000000                3.000000                3.000000
                 Germany                   India                   Japan Korea (Republic of)                   Mexico
                3.000000                3.125000                3.384615                2.666667                3.445833
                 Morocco                 Myanmar             Netherlands               Nicaragua                Pakistan
                3.000000                3.000000                4.000000                3.000000                3.333333
                    Peru             Philippines                  Poland      Russian Federation                   Spain
                3.000000                2.000000                3.000000                3.500000                3.000000
                Thailand                 Tunisia                  Turkey                 Ukraine          United Kingdom
                2.000000                4.000000                3.000000                3.000000                5.000000
United States of America
                5.000000
> print("mean english scores for students from different countries:")
[1] "mean english scores for students from different countries:"
> tapply(std1$english.grade, std1$nationality, mean)
              Bangladesh                  Brazil                  Canada                   Chile                   China
                3.900000                3.577778                3.657143                3.700000                3.253846
                Colombia                    Cuba      Dominican Republic                   Egypt             El Salvador
                3.540000                3.900000                3.400000                1.500000                3.300000
                 Germany                   India                   Japan Korea (Republic of)                   Mexico
                3.200000                3.000000                3.330769                3.533333                3.358333
                 Morocco                 Myanmar             Netherlands               Nicaragua                Pakistan
                2.400000                3.100000                2.500000                3.300000                3.400000
                    Peru             Philippines                  Poland      Russian Federation                   Spain
                3.700000                3.900000                3.900000                3.550000                3.450000
                Thailand                 Tunisia                  Turkey                 Ukraine          United Kingdom
                3.300000                3.500000                2.600000                3.100000                3.400000
United States of America
                3.379275
```

#importing 2nd csv

```
> std2<-read.csv("https://raw.githubusercontent.com/srpayd/R-Analysis/master/StudentsPerformance.csv")
```

OUTPUT:

std1	307 obs. of 15 variables
std2	1000 obs. of 8 variables

#spliting to get only 307 rows

```
std2<-std2[0:307,0:7]
```

OUTPUT:

std1	307 obs. of 15 variables
std2	307 obs. of 7 variables

#adding new column

```
> std2$id<-c(0:306)
```

OUTPUT:

std2	307 obs. of 8 variables

#renaming a column

```
> std2 <- std2 %>%
+   rename("listening.scores"="math.score")
```

#deleting a column

```
> std2<-subset(std2,select=-c(gender))
```

OUTPUT:

std2	307 obs. of 7 variables

#using merge to combine by id

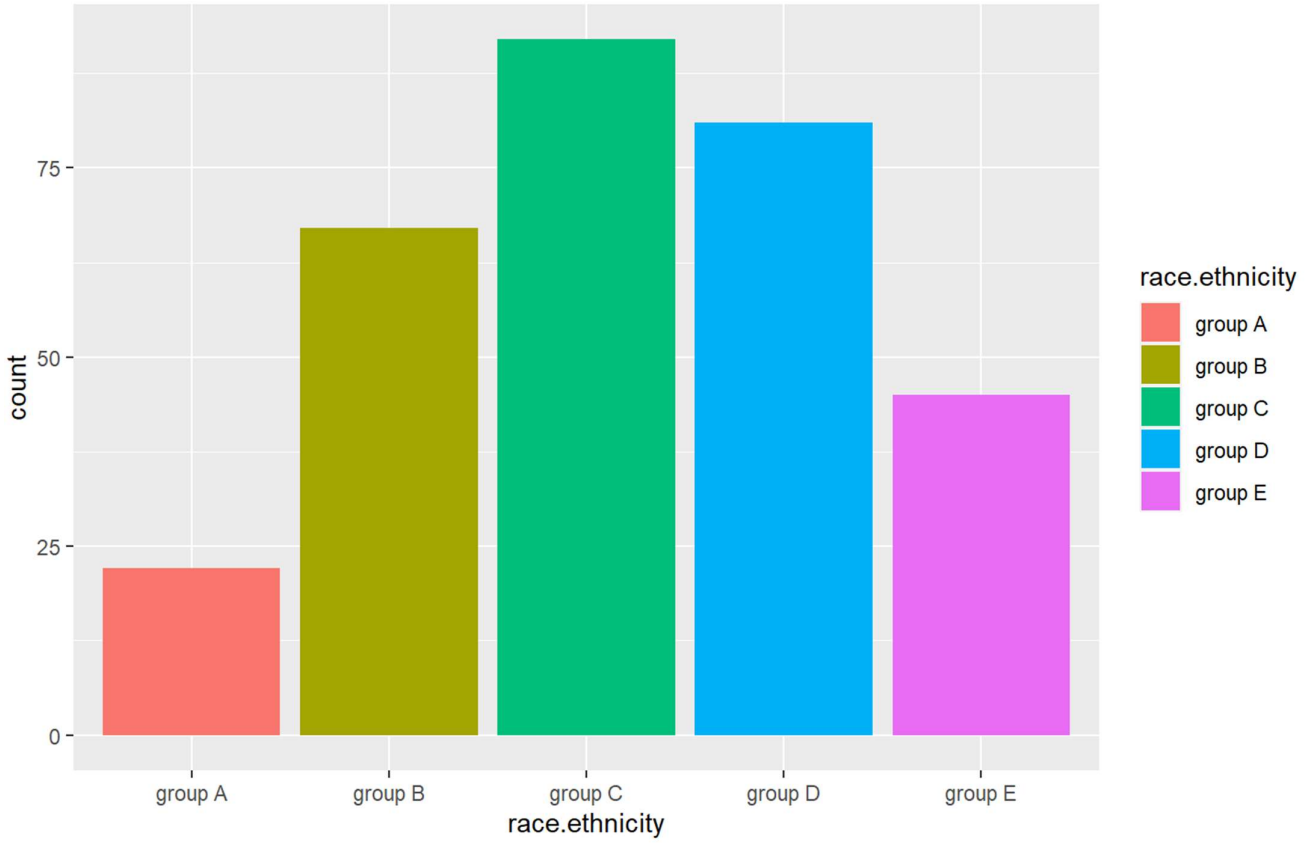
```
> std_combined<-merge(std1,std2,by="id")
```

OUTPUT:

std_combined	307 obs. of 21 variables
std1	307 obs. of 15 variables
std2	307 obs. of 7 variables

```
#bar plot with colors.
ggplot(std_combined,aes(race.ethnicity,fill = race.ethnicity)) +
  geom_bar()
```

OUTPUT:




```
>head(std_combined):
```

Output:

	id	name	nationality	city	latitude	longitude	gender	age	english.grade	math.grade	sciences.grade	language.grade	portfolio.rating	coverletter.rating
1	0	Kiana Lor	China	Suzhou	31.31	120.62	F	22	3.5	3.7	3.1	1.0	4	4.0
2	1	Joshua Lonaker	United States of America	Santa Clarita	34.39	-118.54	M	22	2.9	3.2	3.6	5.0	5	4.0
3	2	Dakota Blanco	United States of America	Oakland	37.80	-122.27	F	22	3.9	3.8	3.2	5.0	3	3.0
4	3	Natasha Yarusso	United States of America	Castro Valley	37.69	-122.09	F	20	3.3	2.8	3.2	5.0	5	2.0
5	4	Brooke Cazares	Brazil	São José dos Campos	-23.18	-45.88	F	21	3.7	2.6	3.4	1.0	4	4.0
6	5	Rochelle Johnson	United States of America	Indianapolis	39.77	-86.16	F	21	3.4	3.1	3.7	5.0	2	4.0
7	6	Joey Abreu	China	Shenyang	41.79	123.43	M	22	3.7	3.9	3.6	2.0	5	5.0
8	7	Preston Suarez	Brazil	São Paulo	-23.47	-46.67	M	22	3.8	3.7	3.6	2.0	5	5.0
9	8	Lee Dong	Philippines	Manila	14.60	120.98	F	24	3.9	3.6	3.2	2.0	4	3.0
10	9	Maa'iz al-Dia	Turkey	Istanbul	41.02	28.96	M	22	2.4	2.8	3.8	3.0	5	5.0
11	10	Maja Nicholson	United States of America	Dallas	32.78	-96.80	F	23	3.4	3.5	3.2	5.0	4	2.0
12	11	Sasha Jansen	United States of America	Chicago	41.85	-87.65	F	21	1.7	4.0	3.6	5.0	4	4.0
13	12	Alexander Sherman	United States of America	Omaha	41.26	-95.94	M	20	3.8	3.6	3.9	5.0	4	4.0
14	13	Edgar Sanchez	Mexico	Tijuana	32.53	-117.02	M	23	3.7	3.5	4.0	4.0	4	5.0
15	14	Kolbi Strunk	United States of America	Mission Viejo	33.60	-117.67	M	21	3.2	3.9	3.3	5.0	4	3.0
16	15	Brittany Sath	Japan	Tokyo	35.69	139.75	F	21	4.0	3.9	3.7	3.0	5	3.0
17	16	Meggan Smith	United States of America	Los Angeles	34.05	-118.24	F	21	2.9	3.4	3.6	5.0	2	5.0
18	17	Ericka Arreola	Mexico	Mexico	19.43	-99.14	other	23	3.1	3.5	2.8	4.0	4	3.0
19	18	David Pulc	Canada	Toronto	43.67	-79.42	M	24	3.9	2.9	4.0	5.0	5	4.0
20	19	Kyle Luckey	United States of America	Moreno Valley	33.94	-117.23	M	23	3.7	3.9	3.4	5.0	3	4.0