

Sample 1

#Install the tidyverse R package and read the mpg dataset

```
library(tidyverse)
```

#Store mpg in a variable

```
mpg_new <- mpg
```

Plot the relationship between engine size (displ) and fuel efficiency (hwy)

```
ggplot( data = mpg) + geom_point(mapping = aes(x = displ, y = hwy, color = class))
```

Generate a scatterplot of **hwy** vs **cyl** in the mpg dataset

```
ggplot( data = mpg) + geom_point(mapping = aes(x = hwy, y = cyl, color = class))
```

Compute and display the total number of 4-, 5-, 6- and 8-cylinder vehicles in the dataset.

```
table(mpg$cyl)
```

#Display the total number of 4-, 5-, 6- and 8-cylinder vehicles

```
ggplot(mpg, aes(x = cyl)) + geom_bar(fill = "steelblue") + geom_text(stat = "count", aes(label = ..count..),  
vjust=1.6, color = "white", size=3.5)
```

Compute and display the details (manufacturer, model, etc.) of the most fuel-efficient vehicles in the dataset.

```
n_index <- which.max(mpg$hwy)
```

```
n_index
```

```
mpg[n_index,]
```

Compute and display the details (manufacturer, model, etc.) of the least fuel-efficient vehicles in the dataset.

```
n_index2 <- which.min(mpg$hwy)
```

```
n_index2
```

```
mpg[n_index2,]
```

Sample 2

#Loading the tidyverse and ggplot 2 libraries

```
library(tidyverse)
```

```
library(ggplot2)
```

#Reading and loading the mpg dataset to a variable

```
workingdata = mpg
```

#creating a plot for relationship between displ and hwy

#and making chart for cars grouped by number of cylinders

```
ggplot(data = workingdata) + geom_point(mapping = aes(x = displ, y = hwy, color = class))
```

Generate a scatterplot of **hwy** vs **cyl** in the mpg dataset

```
ggplot( data = mpg) + geom_point(mapping = aes(x = cyl, y = hwy, color = class))
```

#Compute the total number of 4-, 5-, 6- and 8-cylinder vehicles

```
table(mpg$cyl)
```

```
ggplot(data = workingdata, aes(x=cyl,color=cyl)) + geom_bar()
```

#grabs the row index for the highest and lowest value in cty and hwy

```
index_max_cty = which.max(workingdata$cty)
```

```
index_max_hwy = which.max(workingdata$hwy)
```

```
index_min_cty = which.min(workingdata$cty)
```

```
index_min_hwy = which.min(workingdata$hwy)
```

```
#prints all the details of the highest and lowest value in cty and hwy
```

```
print(workingdata[index_max_cty,])
```

```
print(workingdata[index_min_cty,])
```

```
print(workingdata[index_max_hwy,])
```

```
print(workingdata[index_min_hwy,])
```