AFRAAZ HUSSAIN
20BDS0374
DATA VISUALISATION AND PRESENTATION
LAB ASSESSMENT 1

QUESTION:

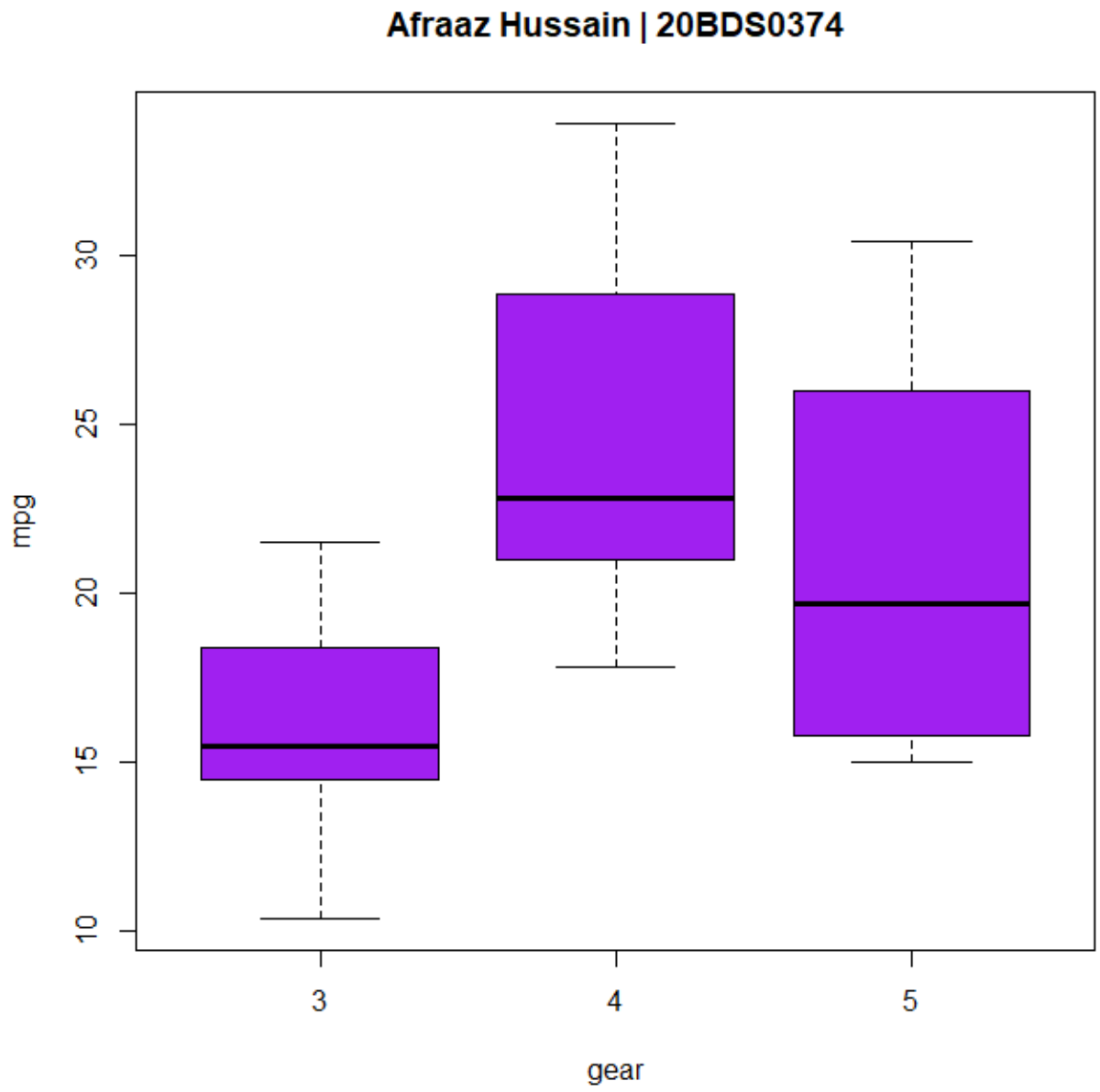
Load the in-build dataset from R and draw various basic plot in R using grid (Horizontal bar plot, Vertical bar plot, box plot, multiple box plot, plot with point an lines etc.,)

CODE:

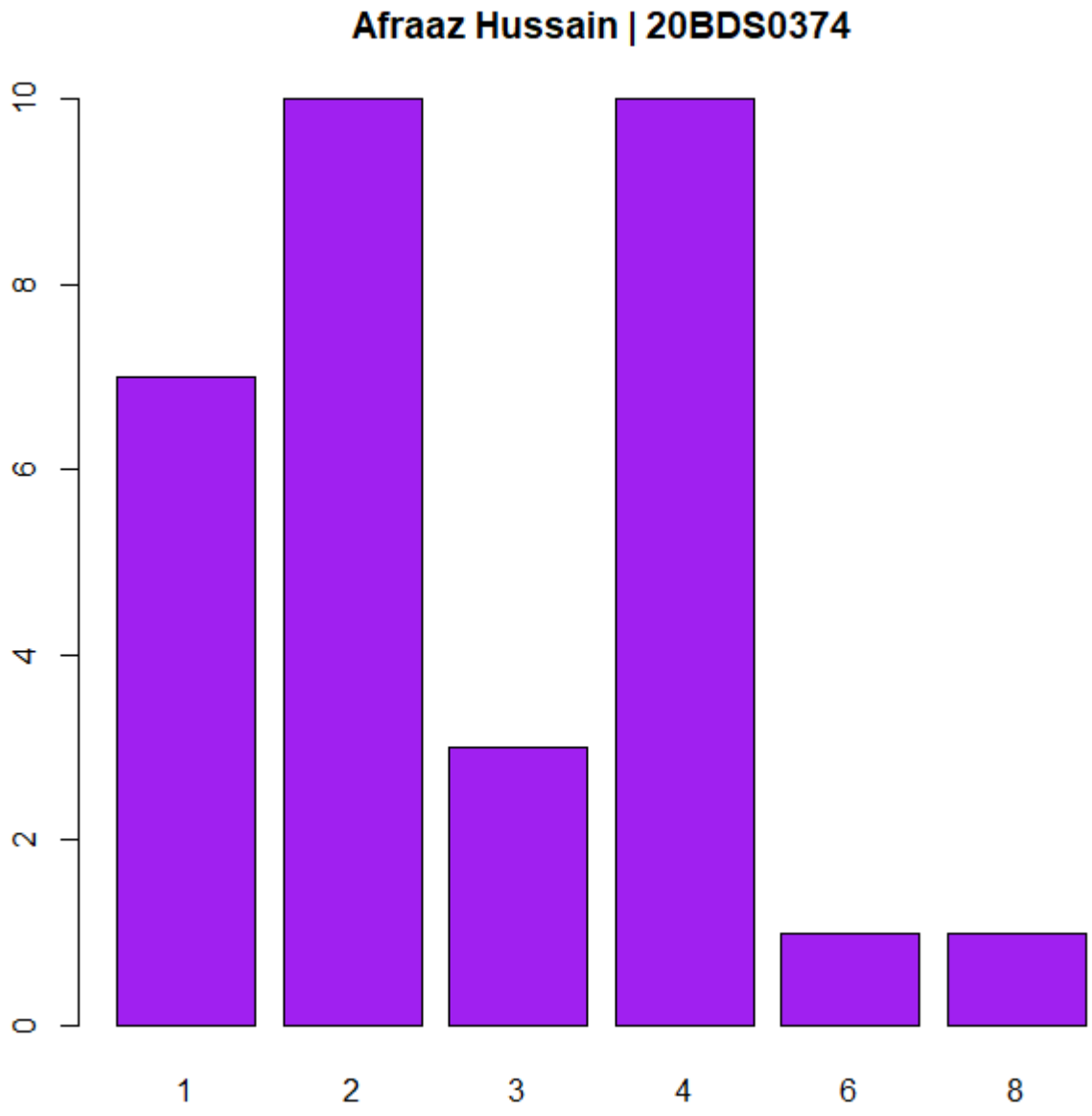
```
library(datasets)


head("mtcars")

summary("mtcars")


boxplot(mpg~gear, data = mtcars, col = "purple")

hist(mtcars$mpg, col = "purple", breaks = 50)

barplot(table(mtcars$carb), col = "purple")

with(mtcars, plot(mpg, qsec))
```

OUTPUT:

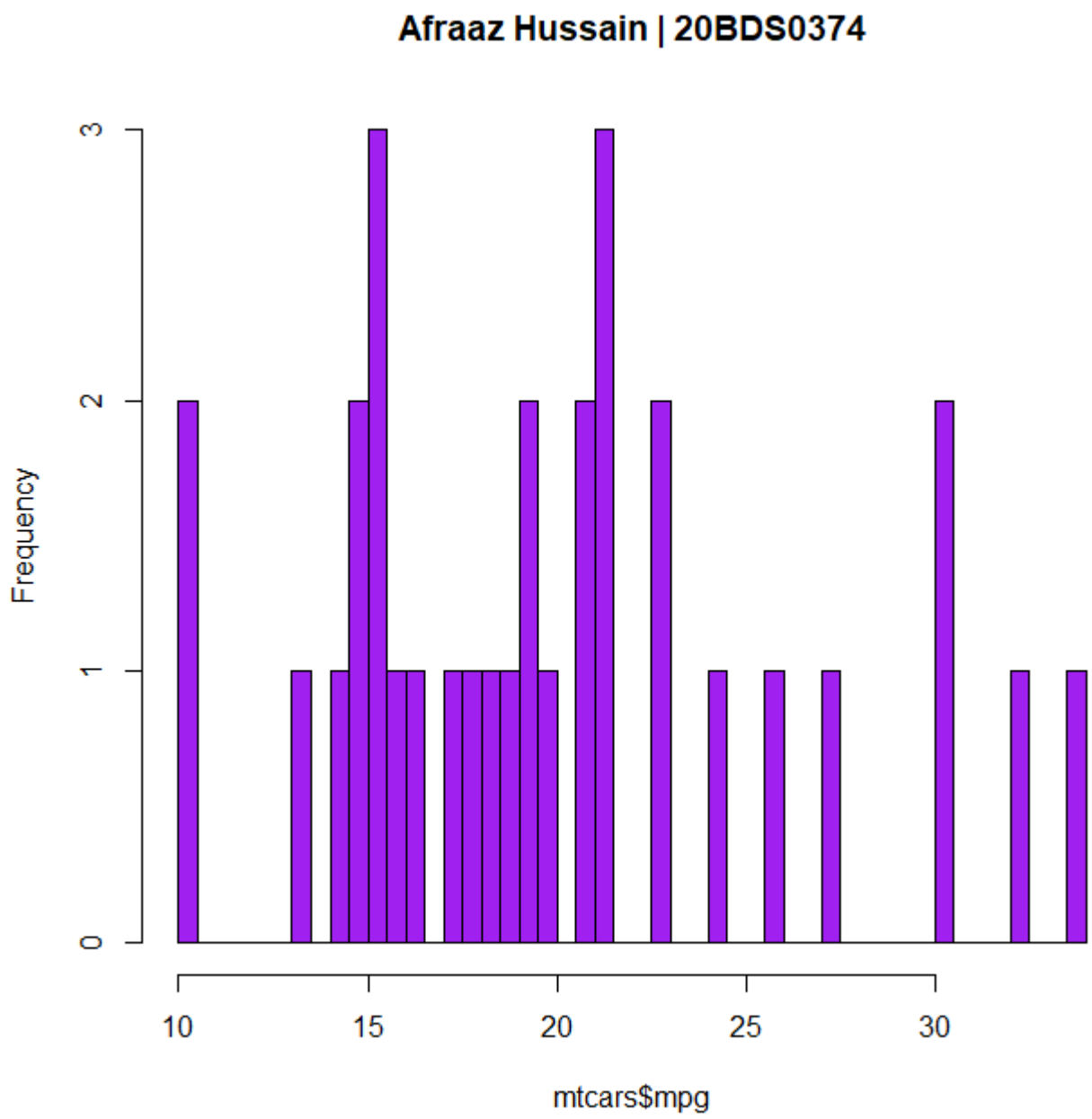
- Boxplot:



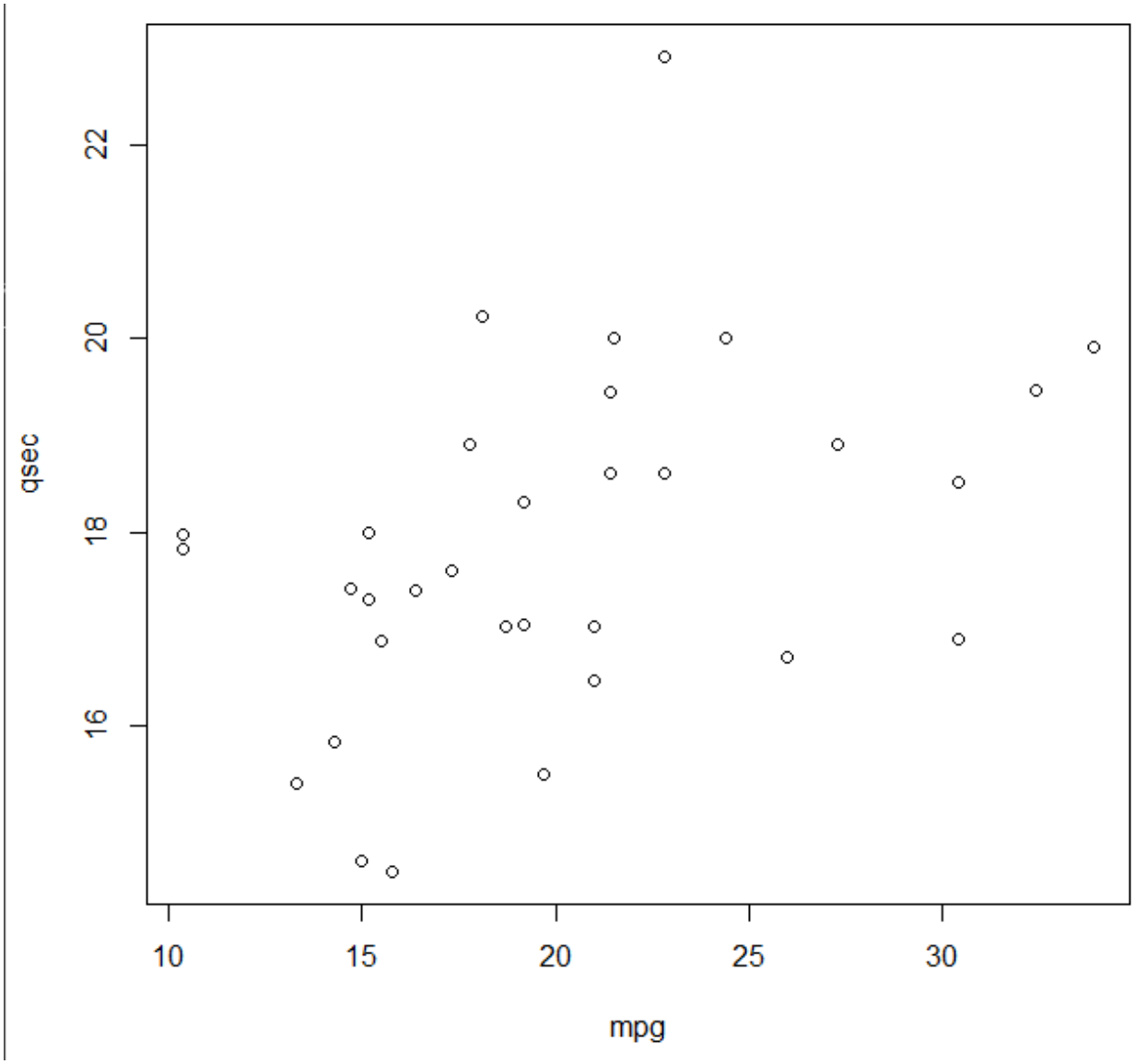
- Bar plot:



- Histogram:


Afraaz Hussain | 20BDS0374
Frequency
0
1
2
3
10
15
20
25
30
mtcars$mpg

- Scatter plot:



AFRAAZ HUSSAIN
20BDS0374
DATA VISUALISATION AND PRESENTATION
LAB ASSESSMENT 1

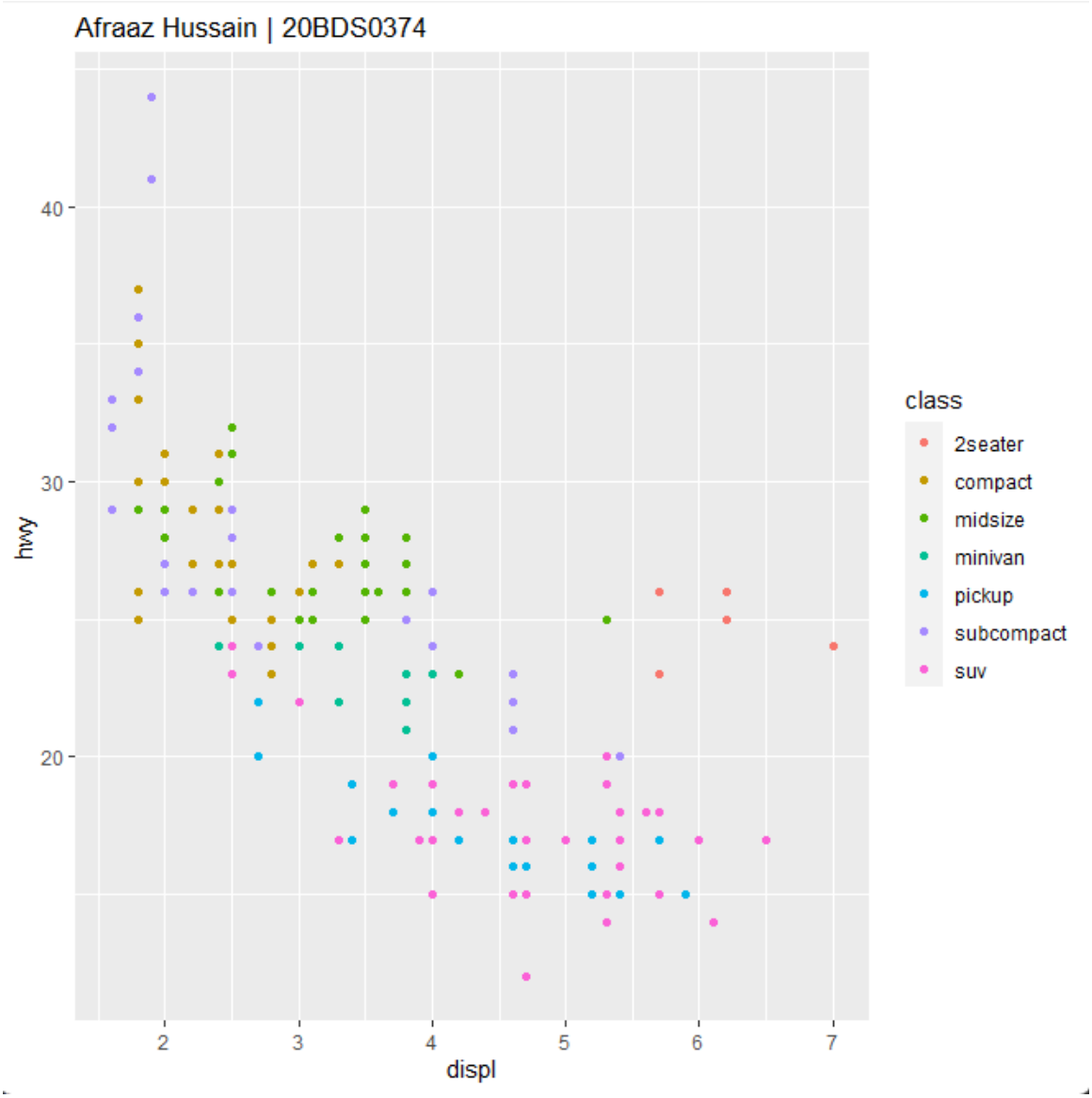
QUESTION:

Load in-build dataset mtcars and visualize data using visualization library ggplot.

CODE:

```
library(tidyverse)

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

OUTPUT:



AFRAAZ HUSSAIN
20BDS0374
DATA VISUALISATION AND PRESENTATION
LAB ASSESSMENT 1

QUESTION:

Load the gapminder dataset and perform statistical analysis using tidyverse and dplyr libraries.

CODE:

```
library(tidyverse)

library(dplyr)

library(gapminder)


gapminder%>% filter(year == 1952)%>% arrange(desc(gdpPercap))

arrange(filter(gapminder, year == 1952), desc(gdpPercap))

gapminder %>% filter(year == 1992, continent == 'Europe')%>% arrange(desc(pop))

gapminder %>% mutate(pop = pop / 1000000)
```

OUTPUT:

- The year is 1952, and the data is arranged in descending order:

```
> gapminder%>% filter(year == 1952)%>% arrange(desc(gdpPercap))
# A tibble: 142 x 6
   country        continent  year lifeExp      pop gdpPercap
   <fct>          <fct>     <int>   <dbl>    <int>     <dbl>
 1 Kuwait         Asia       1952    55.6   160000   108382.
 2 Switzerland    Europe     1952    69.6  4815000    14734.
 3 United States  Americas   1952    68.4 157553000    13990.
 4 Canada         Americas   1952    68.8  14785584    11367.
 5 New Zealand    Oceania    1952    69.4   1994794    10557.
 6 Norway         Europe     1952    72.7   3327728    10095.
 7 Australia      Oceania    1952    69.1   8691212    10040.
 8 United Kingdom Europe     1952    69.2  50430000     9980.
 9 Bahrain        Asia       1952    50.9    120447     9867.
10 Denmark        Europe     1952    70.8   4334000     9692.
# ... with 132 more rows
# i Use `print(n = ...)` to see more rows
```

AFRAAZ HUSSAIN
20BDS0374
DATA VISUALISATION AND PRESENTATION
LAB ASSESSMENT 1

- Using filter, arrange the data with year as 1952 in descending order:

```
> arrange(filter(gapminder, year == 1952), desc(gdpPercap))
# A tibble: 142 x 6
   country        continent  year lifeExp      pop gdpPercap
   <fct>          <fct>     <int>   <dbl>    <int>     <dbl>
 1 Kuwait         Asia       1952    55.6   160000   108382.
 2 Switzerland    Europe     1952    69.6  4815000    14734.
 3 United States  Americas   1952    68.4 157553000    13990.
 4 Canada         Americas   1952    68.8 14785584     11367.
 5 New Zealand    Oceania    1952    69.4  1994794     10557.
 6 Norway         Europe     1952    72.7  3327728     10095.
 7 Australia      Oceania    1952    69.1  8691212     10040.
 8 United Kingdom Europe     1952    69.2 50430000      9980.
 9 Bahrain        Asia       1952    50.9   120447      9867.
10 Denmark        Europe     1952    70.8  4334000      9692.
# ... with 132 more rows
# i Use `print(n = ...)` to see more rows
```

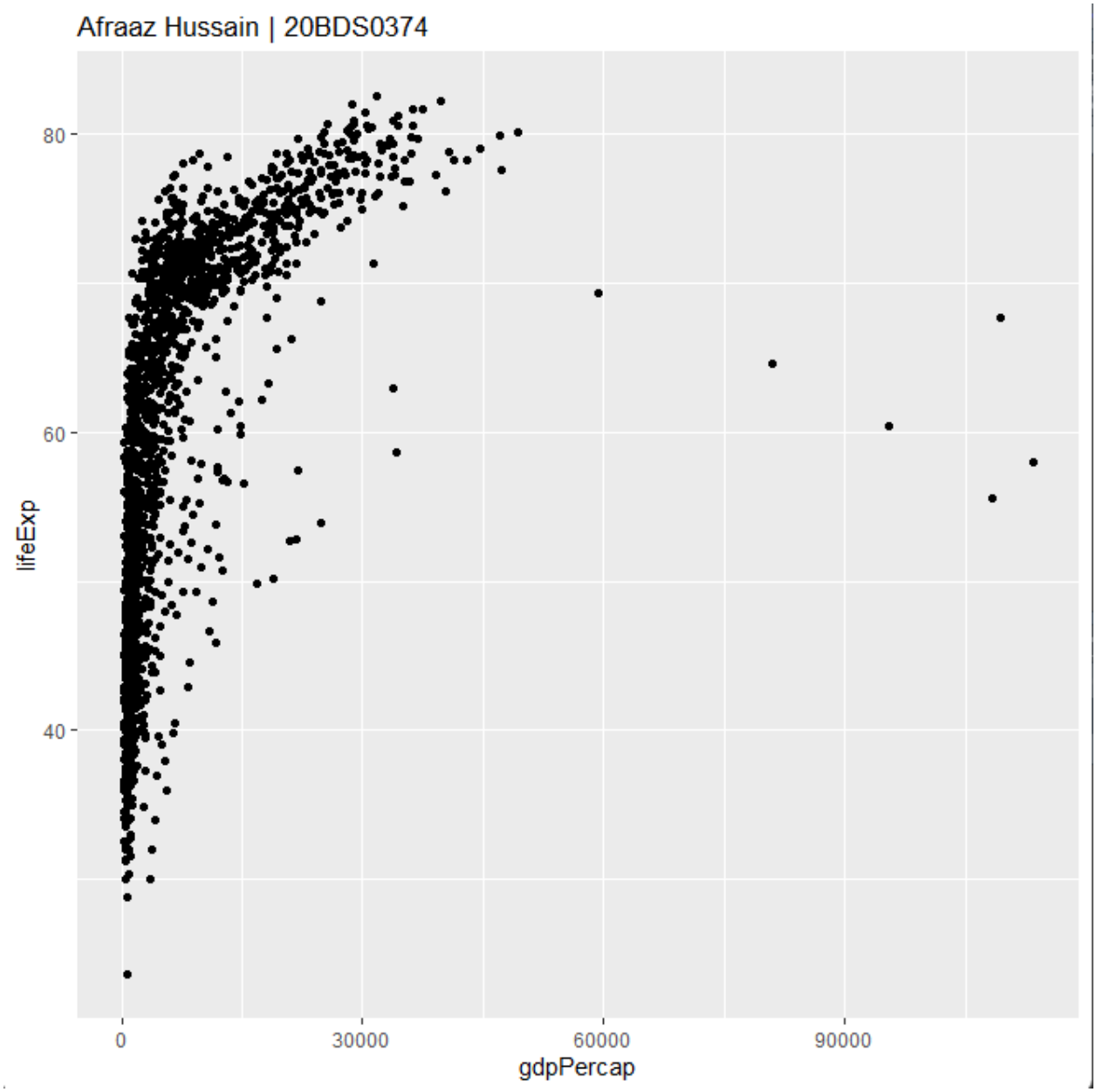
- Year is 1992 and continent is Europe:

```
> gapminder %>% filter(year == 1992, continent == 'Europe')%>% arrange(desc(pop))
# A tibble: 30 x 6
   country        continent  year lifeExp      pop gdpPercap
   <fct>          <fct>     <int>   <dbl>    <int>     <dbl>
 1 Germany        Europe     1992    76.1 80597764     26505.
 2 Turkey         Europe     1992    66.1 58179144      5678.
 3 United Kingdom Europe     1992    76.4 57866349     22705.
 4 France         Europe     1992    77.5 57374179     24704.
 5 Italy          Europe     1992    77.4 56840847     22014.
 6 Spain          Europe     1992    77.6 39549438     18603.
 7 Poland         Europe     1992    71.0 38370697      7739.
 8 Romania        Europe     1992    69.4 22797027      6598.
 9 Netherlands    Europe     1992    77.4 15174244     26791.
10 Hungary        Europe     1992    69.2 10348684     10536.
# ... with 20 more rows
# i Use `print(n = ...)` to see more rows
```

- Mutate the population:

```
> gapminder %>% mutate(pop = pop / 1000000)
# A tibble: 1,704 x 6
   country     continent  year lifeExp   pop gdpPercap
   <fct>       <fct>     <int>   <dbl> <dbl>     <dbl>
 1 Afghanistan Asia       1952    28.8  8.43      779.
 2 Afghanistan Asia       1957    30.3  9.24      821.
 3 Afghanistan Asia       1962    32.0 10.3       853.
 4 Afghanistan Asia       1967    34.0 11.5       836.
 5 Afghanistan Asia       1972    36.1 13.1       740.
 6 Afghanistan Asia       1977    38.4 14.9       786.
 7 Afghanistan Asia       1982    39.9 12.9       978.
 8 Afghanistan Asia       1987    40.8 13.9       852.
 9 Afghanistan Asia       1992    41.7 16.3       649.
10 Afghanistan Asia       1997    41.8 22.2       635.
# ... with 1,694 more rows
# i Use `print(n = ...)` to see more rows
```

- Plot the dataset using ggplot:



AFRAAZ HUSSAIN
20BDS0374
DATA VISUALISATION AND PRESENTATION
LAB ASSESSMENT 1

QUESTION:

Using RClolorBrewer visualize mpg data.

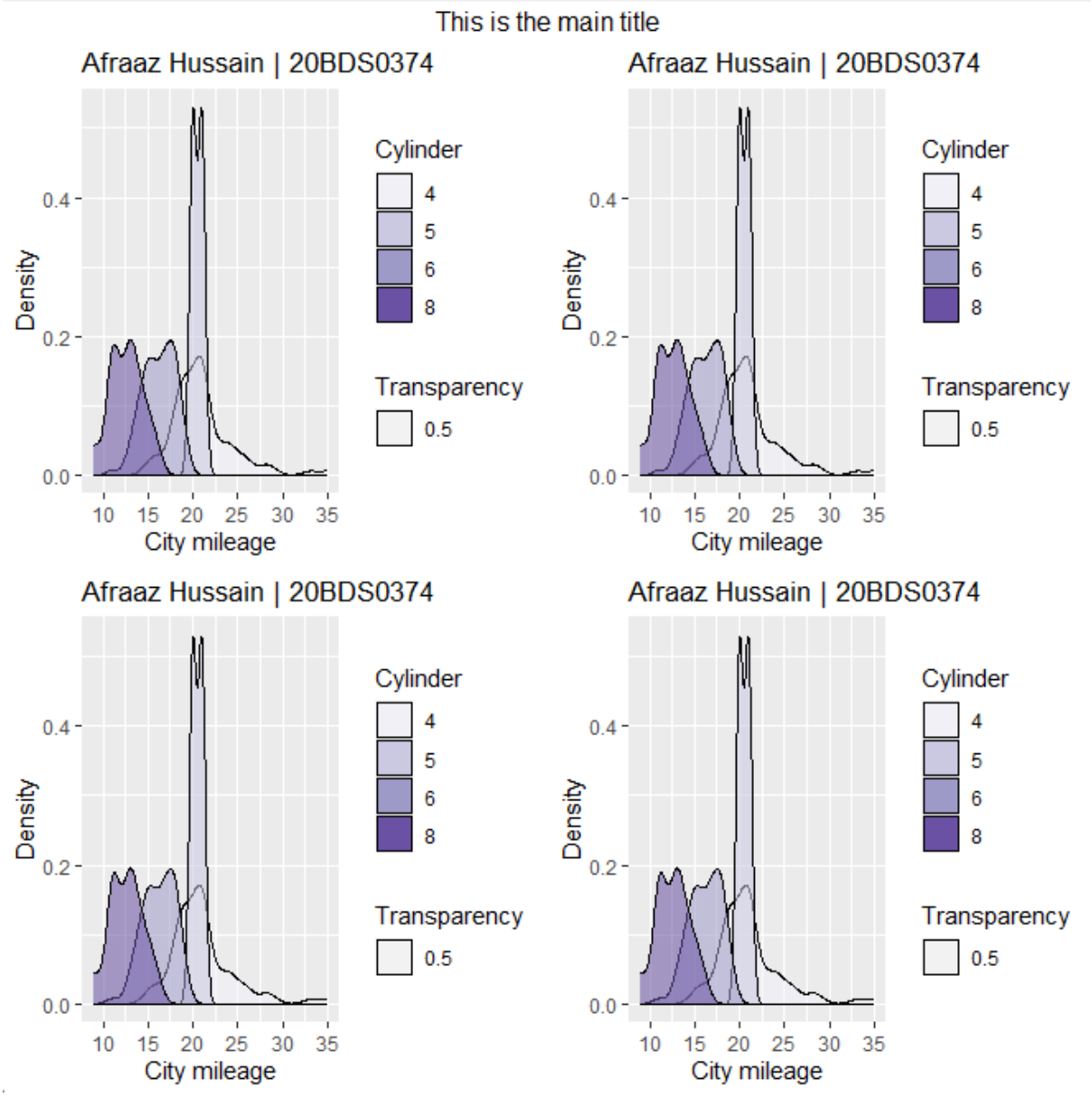
CODE:

```
#Color visualisation using 'RColorBrewer' library
library(RColorBrewer)
library(ggplot2)
library(viridis)
library(datasets)

#The 'View' function is used for viewing a dataset.
#We will now do color visualisation on 'mpg' dataset
View(mpg)

#Here is a density plot on the city
#Use the 'str' function to view the dataset in a structured form
ggplot(data = mpg, aes(x = cty)) + geom_density()
str(mpg)
#Use the 'factor' argument for factoring in different types in a column
#Use the 'alpha' argument for transparency. This argument is not one of 'factor', but outside it
#The 'labs' function is used for assigning labels
#The 'scale_fill_brewer' function is for switching the color palettes of the density graph
p1 = p2 = p3 = p4 = ggplot(data = mpg, aes(x = cty)) +
  geom_density(aes(fill = factor(mpg$cyl), alpha = 0.5)) +
  labs(title = "Afraaz Hussain | 20BDS0374", x = "City mileage", y = "Density", fill = "Cylinder", alpha =
"Transparency") +
  scale_fill_brewer(palette = "Purples")

#This library is for a grid like structure
library(gridExtra)
```

#Now, you can show multiple plots in a single image. Here's how...

grid.arrange(p1, p2, p3, p4, nrow = 2, top = "This is the main title")

OUTPUT:



AFRAAZ HUSSAIN
20BDS0374
DATA VISUALISATION AND PRESENTATION
LAB ASSESSMENT 1

QUESTION:

Load USArrests in-build dataset and correlate in the maps with anyone fields. Display the maps using colormapping.

CODE:

```
#Now, we will import a map and color-map it

library(maps)

library(dplyr)


arrest <- USArrests

View(arrest)


#When viewed, it only has 4 columns. So, we will add a new column with the states in it

#We are converting it to lower-case words as we need to match it to the states in the map's state column

arrest$region = tolower(rownames(arrest))

View(arrest)


#The 'map_data' function contains the longitude, latitude, etc to create a map of US

View(map_data("state"))


#Now, we will join this data with the 'arrest' dataset

statesMap <- map_data("state")

arrestMap <- left_join(statesMap, arrest, by = "region")

View(arrestMap)


#Here, the 'color' argument is for the border of each state

#The 'scale_fill_virdis_c' function is from the 'Viridis' library. It is for applying gradient according to the value

ggplot(data = arrestMap, aes(x = long, y = lat, group = group)) +

  geom_polygon(aes(fill = Assault),
```

```
        color = "white") +
scale_fill_viridis_c(option = "D", direction = 1)
```

OUTPUT: