R Notebook

The following is your first chunk to start with. Remember, you can add chunks using the menu above (Insert -> R) or using the keyboard shortcut Ctrl+Alt+I. A good practice is to use different code chunks to answer different questions. You can delete this comment if you like.

Other useful keyboard shortcuts include Alt- for the assignment operator, and Ctrl+Shift+M for the pipe operator. You can delete these reminders if you don't want them in your report.

Question 2) A.

```
setwd("C:/") #Don't forget to set your working directory before you start!

library("tidyverse")

## Warning: package 'tidyverse' was built under R version 3.6.2

## -- Attaching packages ----------------------------------------------------
-----------------------------------------------------------------------------
---- tidyverse 1.3.0 --

## v ggplot2 3.2.1     v purrr   0.3.3
## v tibble  2.1.3     v dplyr   0.8.3
## v tidyr   1.0.0     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.4.0

## Warning: package 'ggplot2' was built under R version 3.6.1

## Warning: package 'tibble' was built under R version 3.6.2

## Warning: package 'tidyr' was built under R version 3.6.2

## Warning: package 'readr' was built under R version 3.6.2

## Warning: package 'purrr' was built under R version 3.6.2

## Warning: package 'dplyr' was built under R version 3.6.1

## Warning: package 'forcats' was built under R version 3.6.2

## -- Conflicts -------------------------------------------------------------
--------------------------------------------------------------------------- t
idyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()

library("tidymodels")
```

```
## Warning: package 'tidymodels' was built under R version 3.6.2

## -- Attaching packages ---------------------------------------------------------
------------------------------------------------------------------------------
--- tidymodels 0.0.3 --

## v broom     0.5.4     v recipes   0.1.9
## v dials     0.0.4     v rsample   0.0.5
## v infer     0.5.1     v yardstick 0.0.4
## v parsnip   0.0.5

## Warning: package 'dials' was built under R version 3.6.2

## Warning: package 'infer' was built under R version 3.6.2

## Warning: package 'parsnip' was built under R version 3.6.2

## Warning: package 'recipes' was built under R version 3.6.2

## Warning: package 'rsample' was built under R version 3.6.2

## Warning: package 'yardstick' was built under R version 3.6.2

## -- Conflicts -----------------------------------------------------------------
--------------------------------------------------------------------------- ti
dymodels_conflicts() --
## x scales::discard() masks purrr::discard()
## x dplyr::filter()   masks stats::filter()
## x recipes::fixed()  masks stringr::fixed()
## x dplyr::lag()      masks stats::lag()
## x dials::margin()   masks ggplot2::margin()
## x yardstick::spec() masks readr::spec()
## x recipes::step()   masks stats::step()

library("plotly")

## Warning: package 'plotly' was built under R version 3.6.2

##
## Attaching package: 'plotly'

## The following object is masked from 'package:ggplot2':
##
##     last_plot

## The following object is masked from 'package:stats':
##
##     filter

## The following object is masked from 'package:graphics':
##
##     layout
```

```
library("skimr")

## Warning: package 'skimr' was built under R version 3.6.2

library("gapminder")

## Warning: package 'gapminder' was built under R version 3.6.2

dfGap <-  gapminder
dfGap

## # A tibble: 1,704 x 6
##    country     continent  year lifeExp      pop gdpPercap
##    <fct>       <fct>     <int>   <dbl>    <int>     <dbl>
##  1 Afghanistan Asia       1952    28.8  8425333      779.
##  2 Afghanistan Asia       1957    30.3  9240934      821.
##  3 Afghanistan Asia       1962    32.0 10267083      853.
##  4 Afghanistan Asia       1967    34.0 11537966      836.
##  5 Afghanistan Asia       1972    36.1 13079460      740.
##  6 Afghanistan Asia       1977    38.4 14880372      786.
##  7 Afghanistan Asia       1982    39.9 12881816      978.
##  8 Afghanistan Asia       1987    40.8 13867957      852.
##  9 Afghanistan Asia       1992    41.7 16317921      649.
## 10 Afghanistan Asia       1997    41.8 22227415      635.
## # ... with 1,694 more rows
```

Question 3) A.

```
skim(dfGap)
```

Data summary

Name	dfGap
Number of rows	1704
Number of columns	6
_______________________	
Column type frequency:	
factor	2
numeric	4
________________________	
Group variables	None

Variable type: factor

skim_variable	n_missing	complete_rate	ordered	n_unique	top_counts
country	0	1	FALSE	142	Afg: 12, Alb: 12, Alg: 12, Ang: 12

continent	0	1	FALSE	5	Afr: 624, Asi: 396, Eur: 360, Ame: 300

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
year	0	1	1979.50	17.27	1952.00	1965.75	1979.50	1993.25	2007.0	▇▆▆▆▇
lifeExp	0	1	59.47	12.92	23.60	48.20	60.71	70.85	82.6	▁▇▇▇▇
pop	0	1	29601212.32	106157896.74	60011.00	2793664.00	7023595.50	19585221.75	1318683096.0	▇▁▁▁▁
gdpPercap	0	1	7215.33	9857.45	241.17	1202.06	3531.85	9325.46	113523.1	▇▁▁▁▁

Question 3) B.

```
dfGap %>% arrange(desc(lifeExp))%>% filter(year == 2007, lifeExp > 81) %>% se
lect(country)
```

```
## # A tibble: 5 x 1
##   country
##   <fct>
## 1 Japan
## 2 Hong Kong, China
## 3 Iceland
## 4 Switzerland
## 5 Australia
```

```
dfGap
```

```
## # A tibble: 1,704 x 6
##    country     continent  year lifeExp      pop gdpPercap
##    <fct>       <fct>     <int>   <dbl>    <int>     <dbl>
##  1 Afghanistan Asia       1952    28.8  8425333      779.
##  2 Afghanistan Asia       1957    30.3  9240934      821.
##  3 Afghanistan Asia       1962    32.0 10267083      853.
##  4 Afghanistan Asia       1967    34.0 11537966      836.
##  5 Afghanistan Asia       1972    36.1 13079460      740.
##  6 Afghanistan Asia       1977    38.4 14880372      786.
##  7 Afghanistan Asia       1982    39.9 12881816      978.
##  8 Afghanistan Asia       1987    40.8 13867957      852.
##  9 Afghanistan Asia       1992    41.7 16317921      649.
## 10 Afghanistan Asia       1997    41.8 22227415      635.
## # ... with 1,694 more rows
```

Question 3) C.

```
newdF <- dfGap %>% mutate(totalGDP = pop * gdpPercap) %>% filter(year == 2007
) %>% arrange(desc(totalGDP))

newdF

## # A tibble: 142 x 7
##    country        continent  year lifeExp        pop gdpPercap totalGDP
##    <fct>          <fct>     <int>   <dbl>      <int>     <dbl>    <dbl>
##  1 United States  Americas   2007    78.2  301139947    42952.  1.29e13
##  2 China          Asia       2007    73.0 1318683096     4959.  6.54e12
##  3 Japan          Asia       2007    82.6  127467972    31656.  4.04e12
##  4 India          Asia       2007    64.7 1110396331     2452.  2.72e12
##  5 Germany        Europe     2007    79.4   82400996    32170.  2.65e12
##  6 United Kingdom Europe     2007    79.4   60776238    33203.  2.02e12
##  7 France         Europe     2007    80.7   61083916    30470.  1.86e12
##  8 Brazil         Americas   2007    72.4  190010647     9066.  1.72e12
##  9 Italy          Europe     2007    80.5   58147733    28570.  1.66e12
## 10 Mexico         Americas   2007    76.2  108700891    11978.  1.30e12
## # ... with 132 more rows

newdF %>%  select("country", "gdpPercap") %>% arrange(desc(gdpPercap))

## # A tibble: 142 x 2
##    country            gdpPercap
##    <fct>                  <dbl>
##  1 Norway                49357.
##  2 Kuwait                47307.
##  3 Singapore             47143.
##  4 United States         42952.
##  5 Ireland               40676.
##  6 Hong Kong, China      39725.
##  7 Switzerland           37506.
##  8 Netherlands           36798.
##  9 Canada                36319.
## 10 Iceland               36181.
## # ... with 132 more rows

newdF

## # A tibble: 142 x 7
##    country        continent  year lifeExp        pop gdpPercap totalGDP
##    <fct>          <fct>     <int>   <dbl>      <int>     <dbl>    <dbl>
##  1 United States  Americas   2007    78.2  301139947    42952.  1.29e13
##  2 China          Asia       2007    73.0 1318683096     4959.  6.54e12
##  3 Japan          Asia       2007    82.6  127467972    31656.  4.04e12
##  4 India          Asia       2007    64.7 1110396331     2452.  2.72e12
##   5 Germany        Europe     2007    79.4  82400996    32170. 2.65e12
##   6 United Kingdom Europe     2007    79.4  60776238    33203. 2.02e12
##   7 France         Europe     2007    80.7  61083916    30470. 1.86e12
##   8 Brazil         Americas   2007    72.4 190010647     9066. 1.72e12
##   9 Italy          Europe     2007    80.5  58147733    28570. 1.66e12
##  10 Mexico         Americas   2007    76.2 108700891    11978. 1.30e12
## # ... with 132 more rows
```

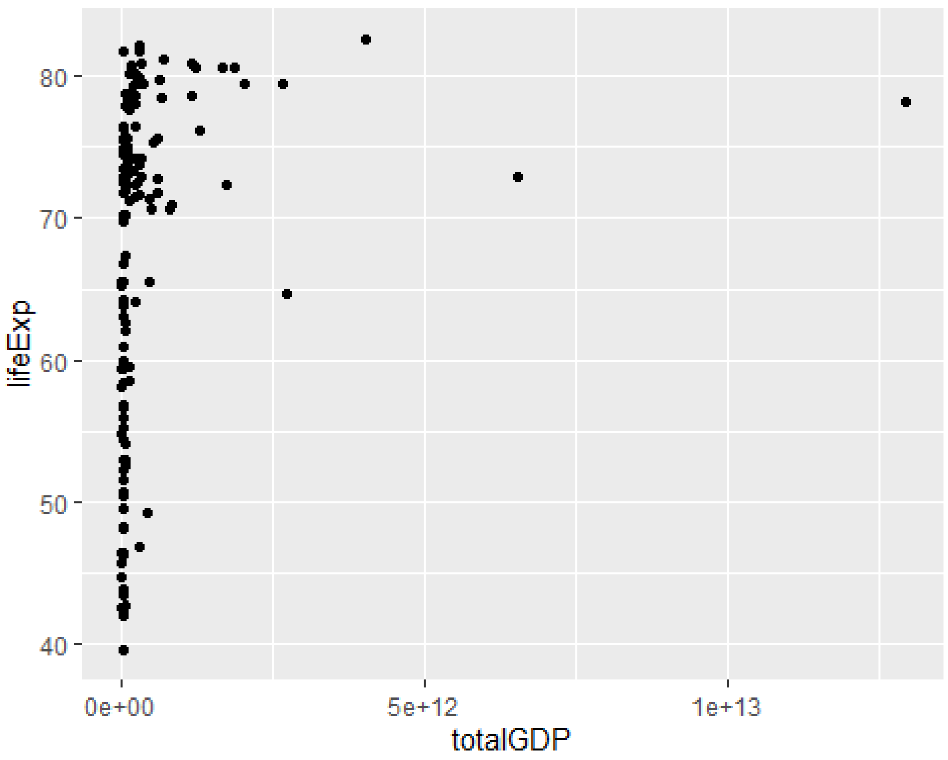
Question 3) D.

```
continents <- dfGap %>% filter(year == 2007) %>%
  group_by(continent) %>% summarise(mdLifeExp = median(lifeExp), mdTotalgdp =
median(gdpPercap)) %>%
  ungroup() %>%
  arrange(desc(mdLifeExp))

continents
```

```
## # A tibble: 5 x 3
##   continent mdLifeExp mdTotalgdp
##   <fct>         <dbl>      <dbl>
## 1 Oceania        80.7     29810.
## 2 Europe         78.6     28054.
## 3 Americas       72.9      8948.
## 4 Asia           72.4      4471.
## 5 Africa         52.9      1452.
```

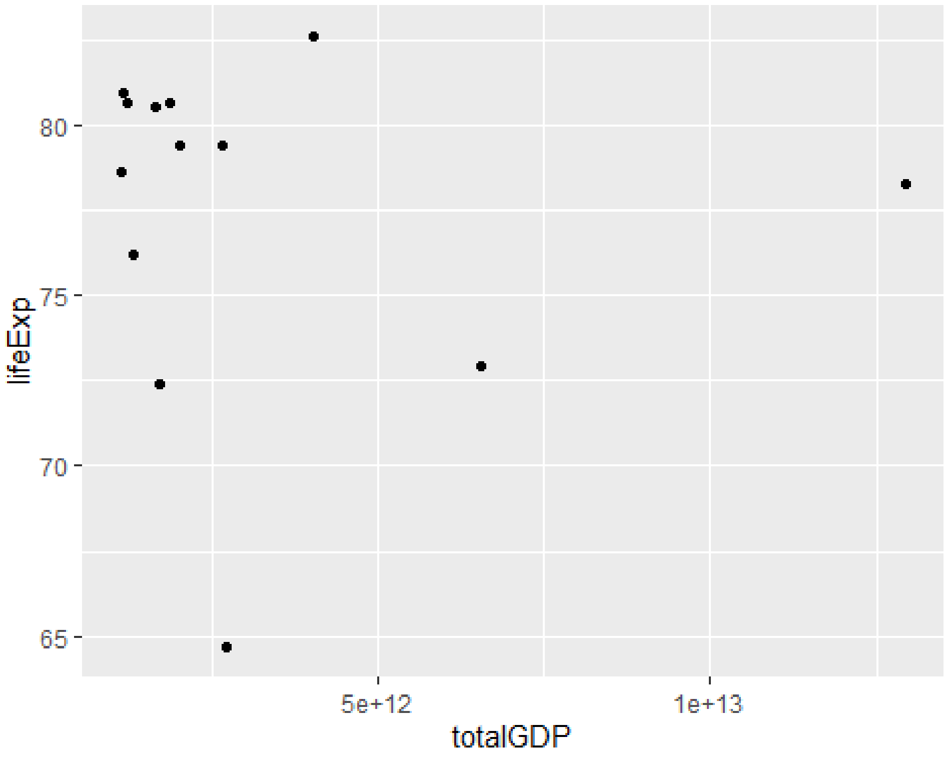
Question 4) A. i)

```
plot1 <- newdF %>%
  ggplot(aes(x=totalGDP ,y=lifeExp)) +
  geom_point()

plot1
```



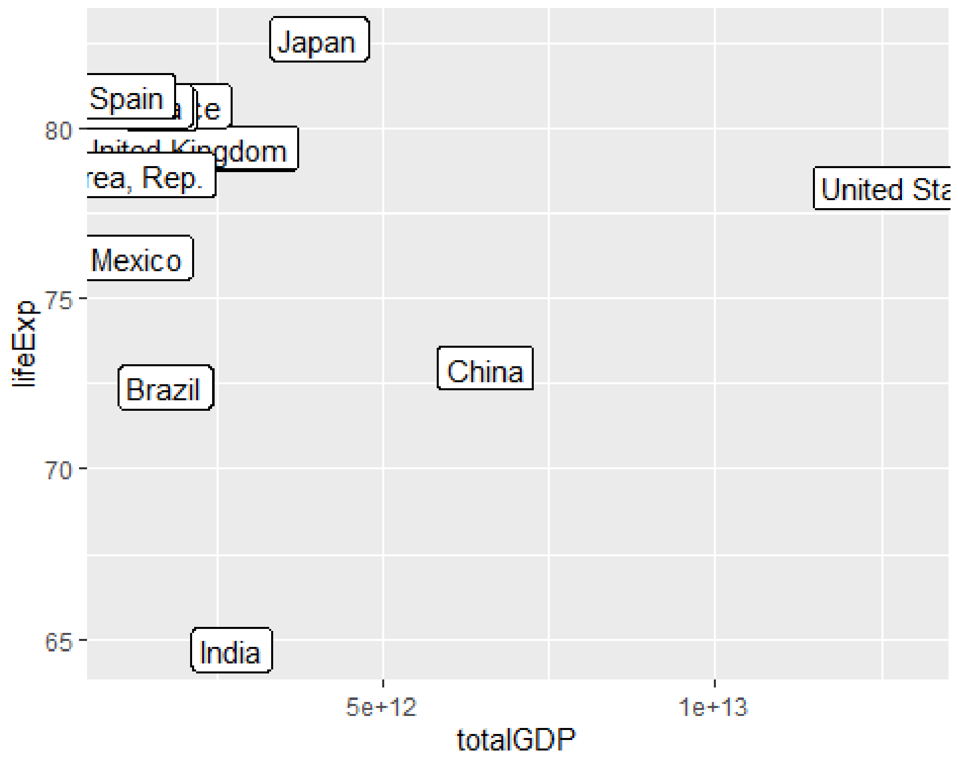
Question 4) A. ii)

```
newdf2 <- newdF %>%  filter(year== 2007, totalGDP > 1e+12)
plot2 <-newdf2 %>%
  ggplot(aes(x=totalGDP ,y=lifeExp)) +
  geom_point()

plot2
```



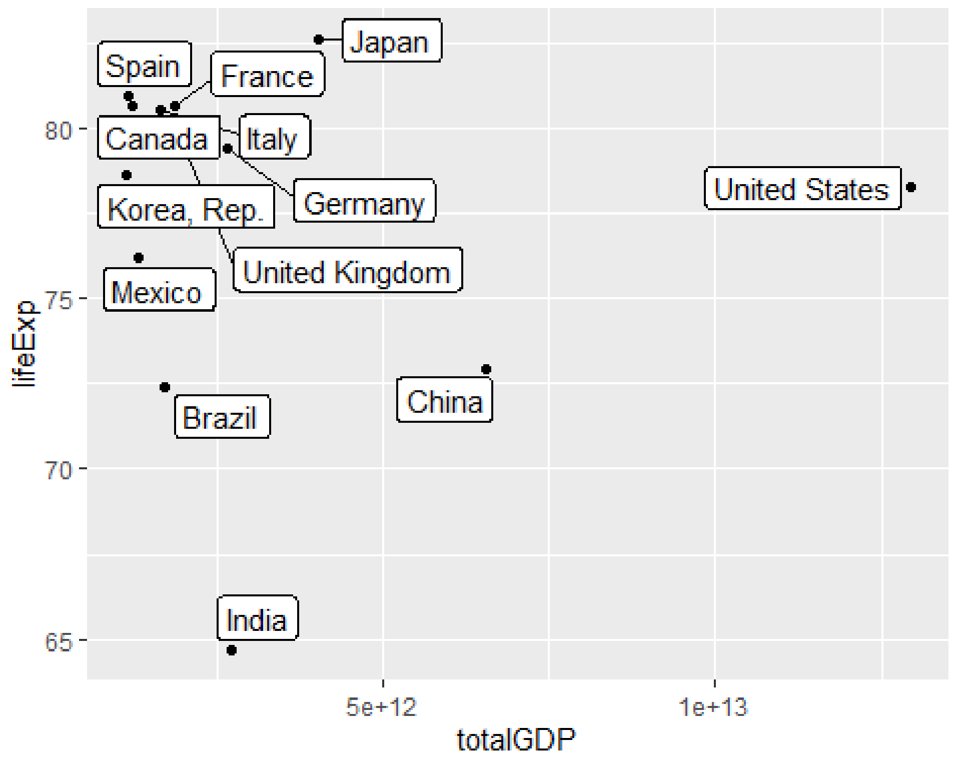
Question 4) A. iii)

```
newdf2 <- newdF %>%  filter(year== 2007, totalGDP > 1e+12)
plot3 <-newdf2 %>%
  ggplot(aes(x=totalGDP ,y=lifeExp)) +
  geom_point() +
  geom_label(aes(x=totalGDP ,y=lifeExp, label= country))

plot3
```



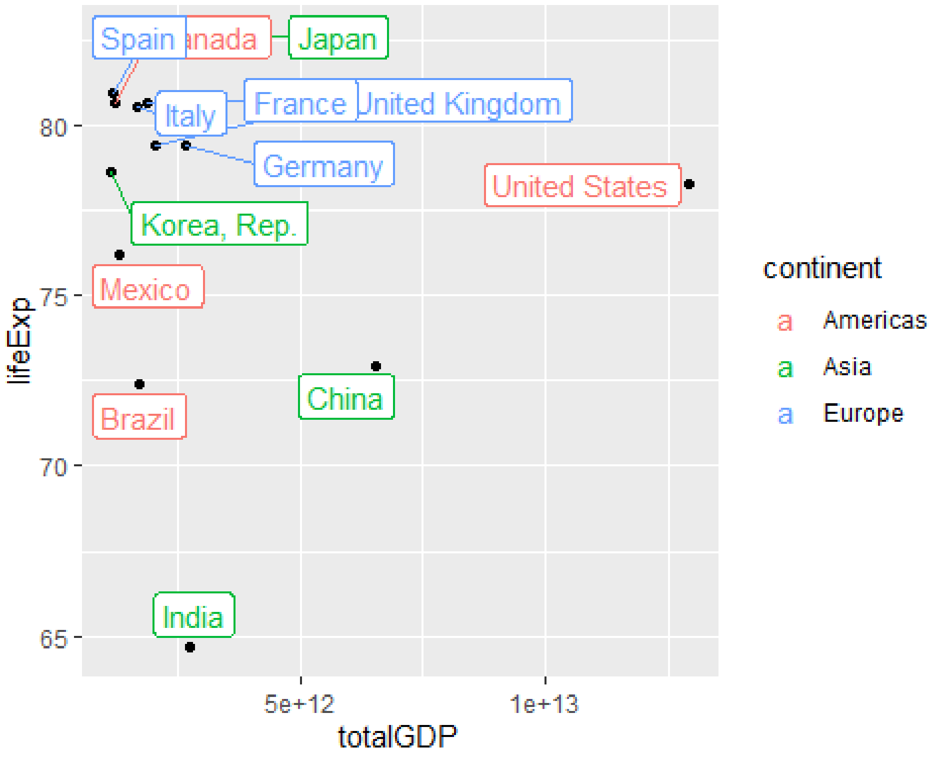
Question 4) A. iv)

```
library(ggrepel)

## Warning: package 'ggrepel' was built under R version 3.6.2

plot4 <-
  newdF %>%
  filter(year== 2007, totalGDP > 1e+12)%>%
  ggplot() +
  geom_point(aes(x=totalGDP ,y=lifeExp)) +
  geom_label_repel(aes(x=totalGDP ,y=lifeExp, label= country,))

plot4
```



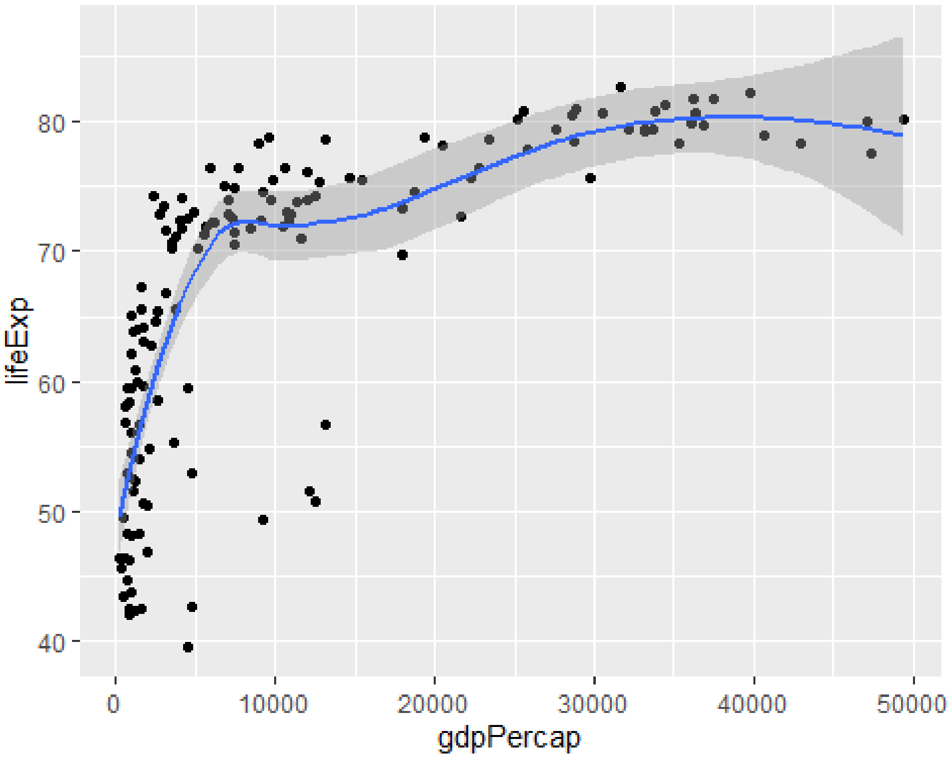
Question 4) A. v)

```
library(ggrepel)

plotx <-
  newdF %>%
  filter(year== 2007, totalGDP > 1e+12)%>%
  ggplot() +
  geom_point(aes(x=totalGDP ,y=lifeExp)) +
  geom_label_repel(aes(x=totalGDP ,y=lifeExp, label= country,color= continent
))

plotx
```

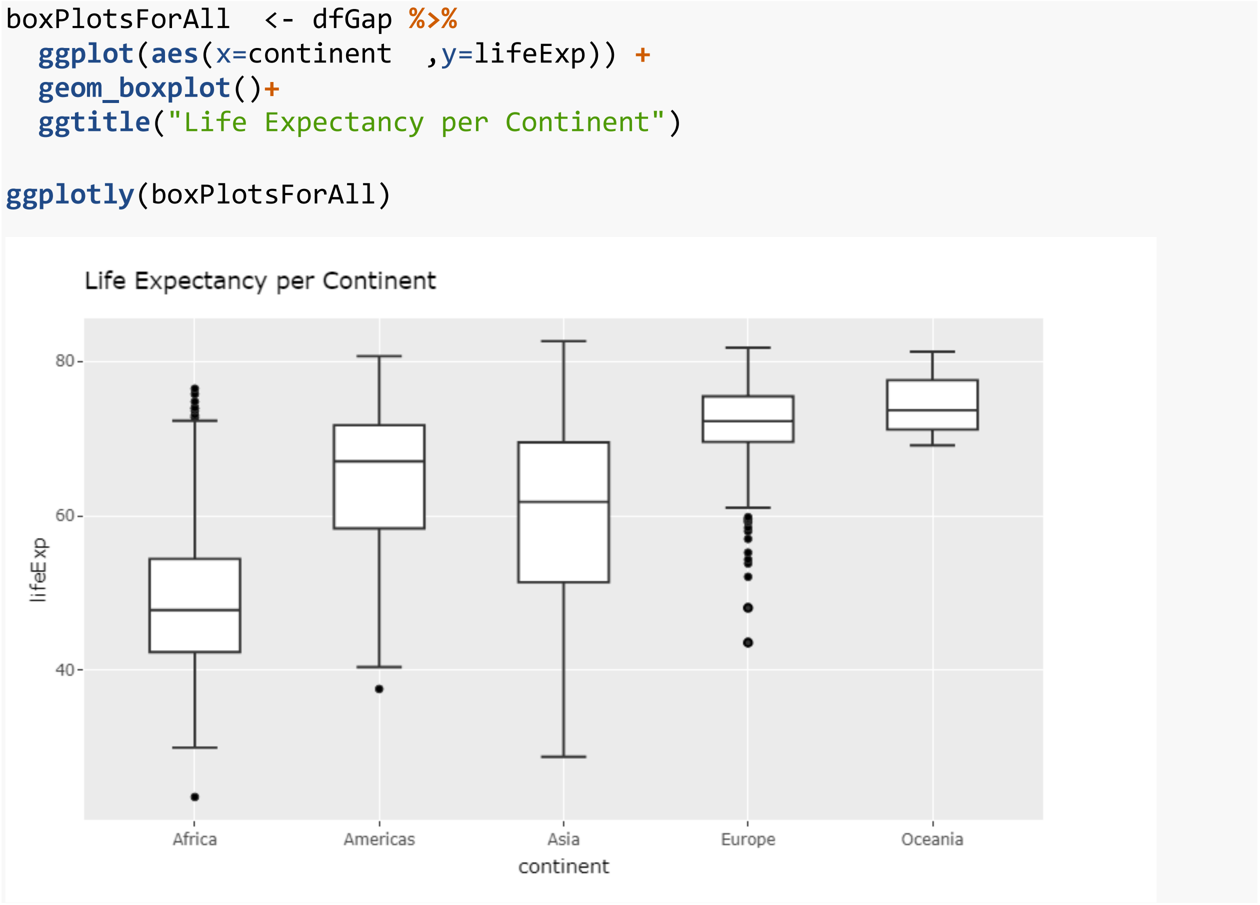


Question 4) B.

```
plot5 <- newdF %>%
  filter(year == 2007) %>%
  ggplot(aes(x=gdpPercap  ,y=lifeExp)) +
  geom_point()+
  geom_smooth()

plot5

## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



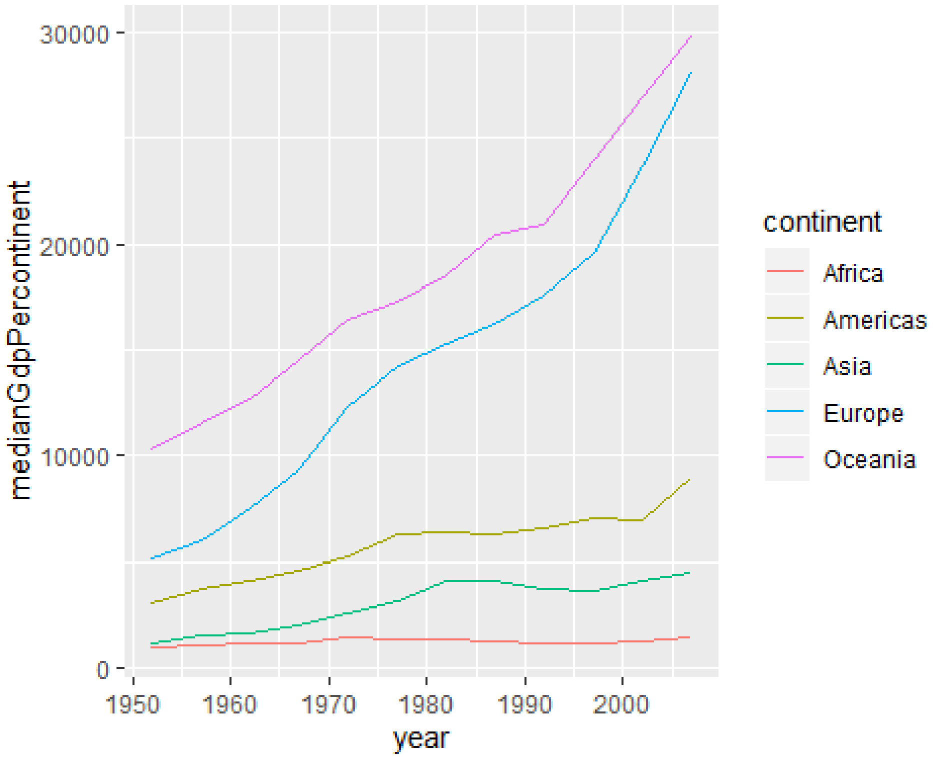
Question 4) C.



```
boxPlotsForAll  <- dfGap %>%
  ggplot(aes(x=continent  ,y=lifeExp)) +
  geom_boxplot()+
  ggtitle("Life Expectancy per Continent")

ggplotly(boxPlotsForAll)
```

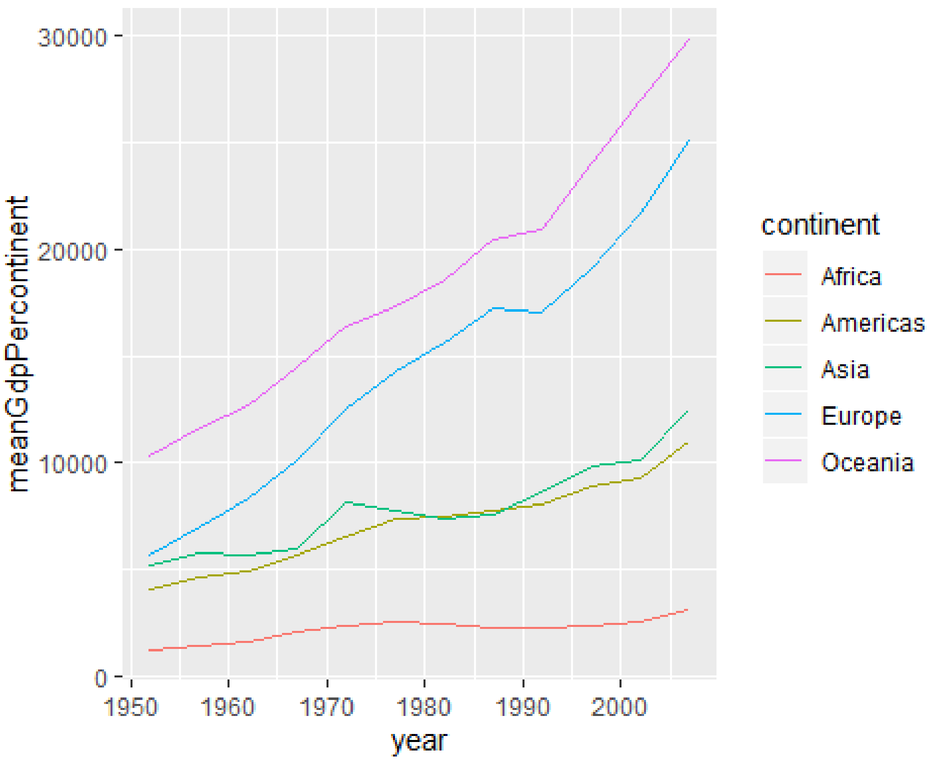
Question 4) D. i)

```
plot6 <- dfGap %>%
  group_by(year, continent) %>%
  mutate(medianGdpPercontinent = median(gdpPercap)) %>%
  ungroup() %>%
  ggplot() +
  geom_line(aes(x= year, y= medianGdpPercontinent, color= continent))


plot6
```



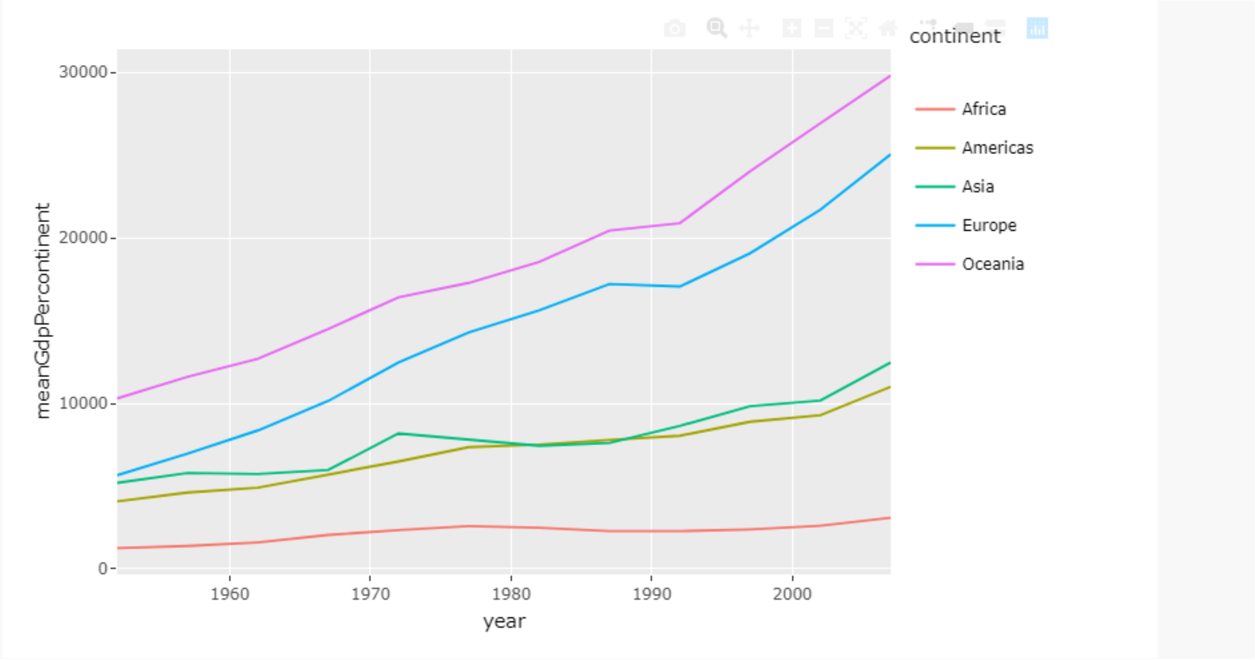
Question 4) D. ii)

```
plot7 <- dfGap %>%
  group_by(year, continent) %>%
  mutate(meanGdpPercontinent = mean(gdpPercap)) %>%
  ungroup() %>%
  ggplot() +
  geom_line(aes(x= year, y= meanGdpPercontinent, color= continent))


plot7
```



Question 4) D. iii)

```
gpdInTime <-
  dfGap %>%
  group_by(year, continent) %>%
  mutate(meanGdpPercontinent = mean(gdpPercap)) %>%
  ungroup() %>%
  ggplot() +
  geom_line(aes(x= year, y= meanGdpPercontinent, color= continent))

ggplotly(gpdInTime)
```


continent
Africa
Americas
Asia
Europe
Oceania
meanGdpPercontinent
30000
20000
10000
0
1960
1970
1980
1990
2000
year