R Notebook

This Exploratory Data analysis is based on the very famous Gapminder Dataset. Dataset link: https://www.gapminder.org/tag/download-data/ Using Basic functionalities available in R, we will explore data and try to resolve some quetsions step by step.

Installing the basic libraries in R

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

library("tidyverse")
```

```
## ── Attaching packages ─────────────────────────────────────── tidyverse
1.3.0 ──

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

## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library("tidymodels")
```

```
## ── Attaching packages ─────────────────────────────────────── tidymodels
0.0.3 ──

## ✓ broom     0.5.3     ✓ recipes   0.1.9
## ✓ dials     0.0.4     ✓ rsample   0.0.5
## ✓ infer     0.5.1     ✓ yardstick 0.0.4
## ✓ parsnip   0.0.5

## ── Conflicts ──────────────────────────────────────── tidymodels_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()
## x recipes::yj_trans() masks scales::yj_trans()
```

```
library("plotly")
```

```
##
## 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")
```

2. Load the data In a new chunk, load the gapminder library, and use this line to create dfGap: dfGap <- gapminder

```
library("gapminder")

dfGap <- gapminder
```

3. Explore the data

a. Use the skim function on the dfGap dataframe to get summary statistics in a nice format. I suggest you use the widest screen possible for the best reading.

```
skim(dfGap)
```

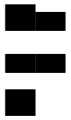
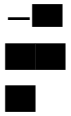
Data summary

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	10615 7896.74	60011.00	2793664.00	7023595.50	19585221.75	13186 83096.0	
gdpPercap	0	1	7215.33	9857.45	241.17	1202.06	3531.85	9325.46	11352 3.1	

3.b.For the year 2007 sort and filter data in descending order of life expectancy.

```
dfGap2007 <-
  filter(dfGap, year == 2007) %>%
  arrange(desc(lifeExp))
dfGap2007
```

```
## # A tibble: 142 x 6
##    country          continent  year lifeExp       pop gdpPercap
##    <fct>            <fct>     <int>   <dbl>     <int>     <dbl>
##  1 Japan            Asia       2007    82.6 127467972    31656.
##  2 Hong Kong, China Asia       2007    82.2   6980412    39725.
##  3 Iceland          Europe     2007    81.8    301931    36181.
##  4 Switzerland      Europe     2007    81.7   7554661    37506.
##  5 Australia        Oceania    2007    81.2  20434176    34435.
##  6 Spain            Europe     2007    80.9  40448191    28821.
##  7 Sweden           Europe     2007    80.9   9031088    33860.
##  8 Israel           Asia       2007    80.7   6426679    25523.
##  9 France           Europe     2007    80.7  61083916    30470.
## 10 Canada           Americas   2007    80.7  33390141    36319.
## # … with 132 more rows
```

3.b.i.What are the names of the countries with a life expectancy over 81?

Japan, Hong Kong, China, Iceland, Switzerland, Australia

3.c. Add a calculated column totalGDP to dfGap showing the total GDP per country, filter the dataframe for 2007, and sort in descending order for totalGDP. If you like, save the new dataframe as a new one for repeated use

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

dfGapTotGDP
```

```
## # 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
```

3.c.i.What are some names of the countries with the top levels of total GDP? United States, China, Japan, India, Germany

3.c.ii.Which ones of these countries overlap with the countries from 3-b? Japan

3.c.iii.What if you selected only the two columns country and gdpPercap and sorted the dataframe in descending order for gdpPercap? Do you observe more of an overlap now? What do you infer from this difference?

```
dfGapTotGDP %>%
  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
```

Canada, Switzerland, Iceland

3.d. Filter dfGap for 2007, group it by continent, and then calculate the median life expectancy and median total GDP (so you need to have totalGDP already).

```
dfGap2007<-
  dfGapTotGDP %>%
  group_by(continent) %>%
  summarize(median_lifeExp = median(lifeExp), medianTotgdp =
median(totalGDP)) %>%
  ungroup() %>%
  arrange(desc(median_lifeExp))

dfGap2007
```

```
## # A tibble: 5 x 3
##   continent median_lifeExp  medianTotgdp
##   <fct>              <dbl>         <dbl>
## 1 Oceania             80.7 403657044512.
## 2 Europe              78.6 230988745548.
## 3 Americas            72.9  65203833292.
## 4 Asia                72.4 164029908950.
## 5 Africa              52.9  13755919229.
```

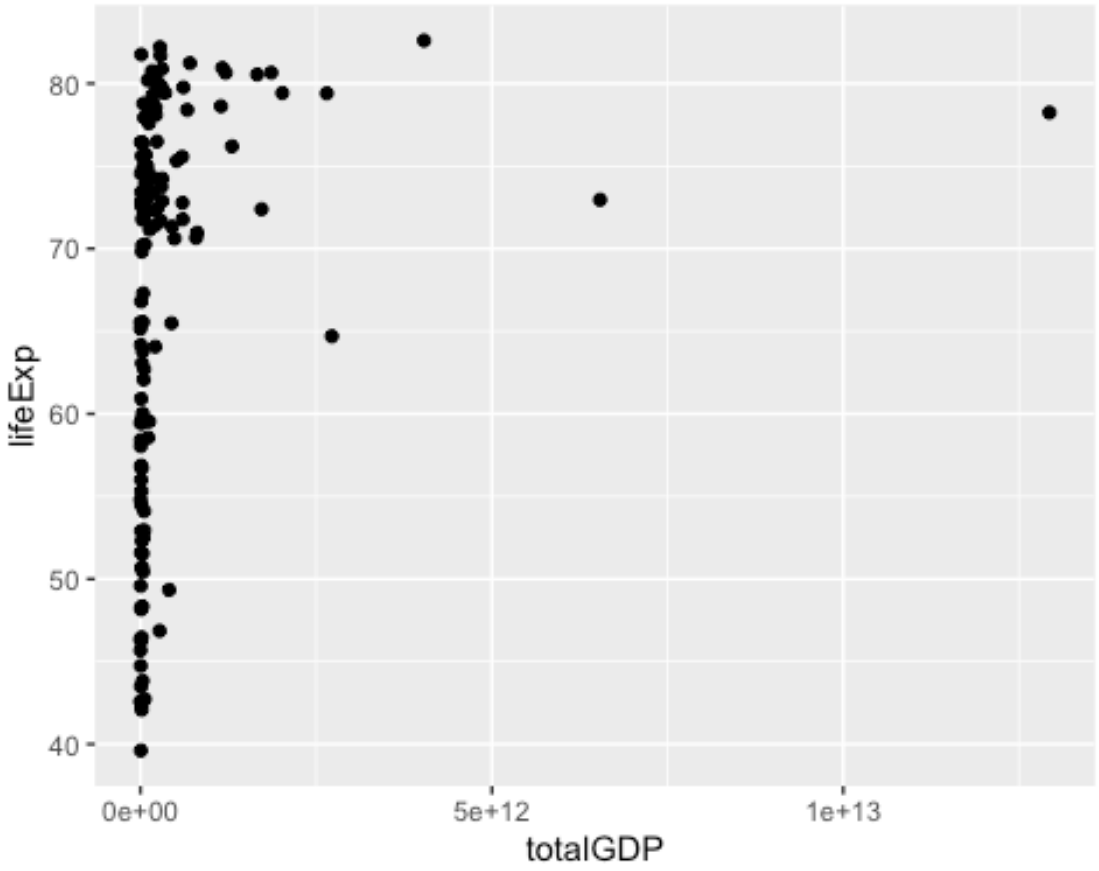
i. What continent has the highest median of life expectancy? Oceania

ii.Does it seem to be correlated with the median total GDP? Yes, Median total GDP and median life exp are positively correlated.

4.a.Visualize the data Now that you have explored the relationship between life expectancy and totalGDP in a table format, let's also visualize it to see a bigger picture.

i.Create a scatter plot to understand the relationship between life expectancy (y-axis) and totalGDP (x-axis) in 2007. Does this plot help?

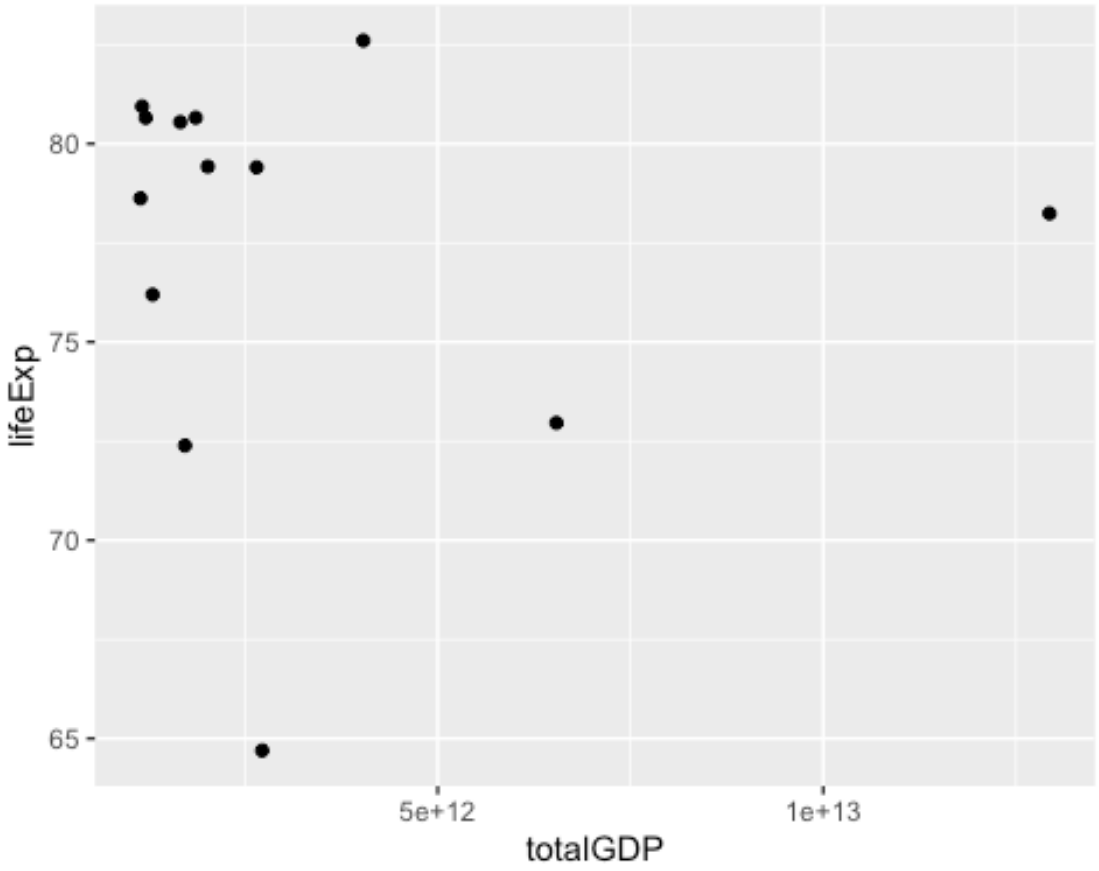
```
dfGapTotGDP %>%
  ggplot(aes(x= totalGDP, y = lifeExp)) + geom_point()
```



Above 70, we are observing more GDP.

ii. Copy the same code, but this time also filter for countries with a totalGDP of over a Trillion (use the scientific notation 1e+12). What about now?

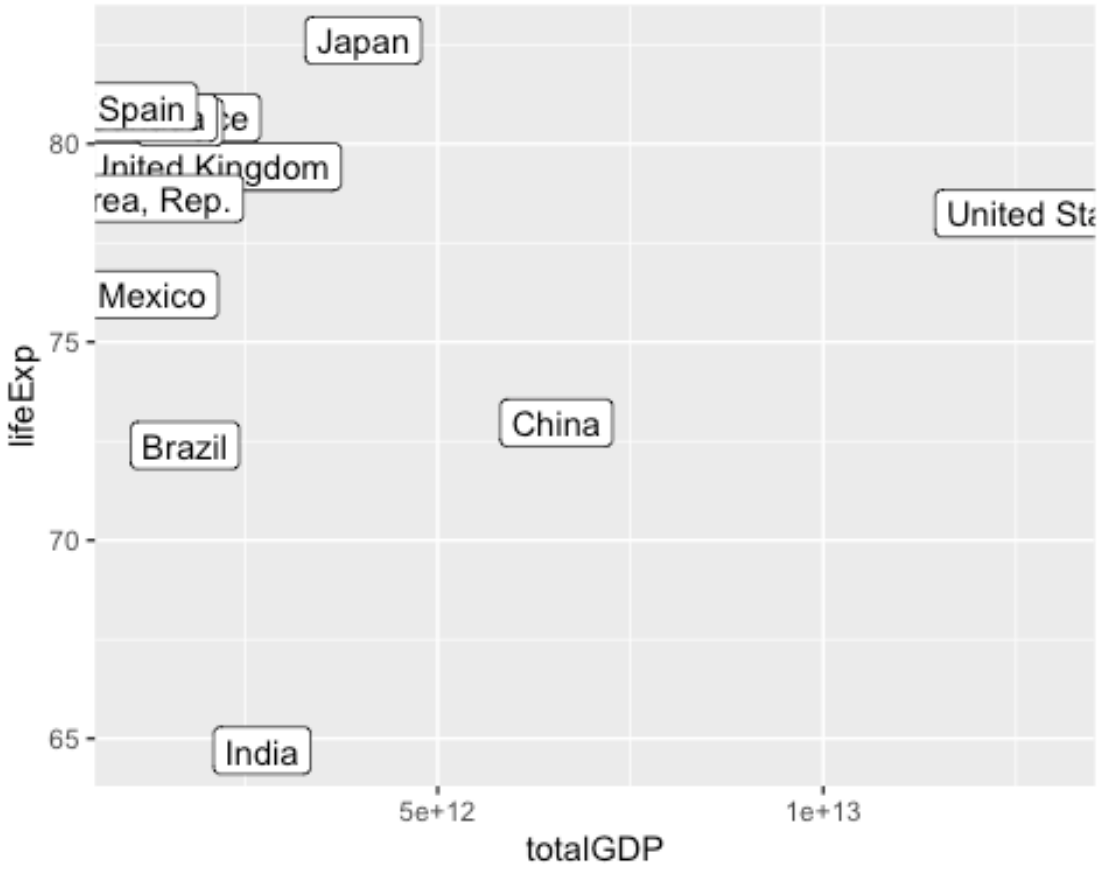
```
dfGapTotGDP %>%
  filter(totalGDP>1e+12) %>%
  ggplot(aes(x= totalGDP, y = lifeExp)) + geom_point()
```



Yes, we get clear understanding with definite scattered points.

iii.Copy the same code, and add labels this time. Do you see a cluster now? What are the names of the countries that are outside of the cluster?

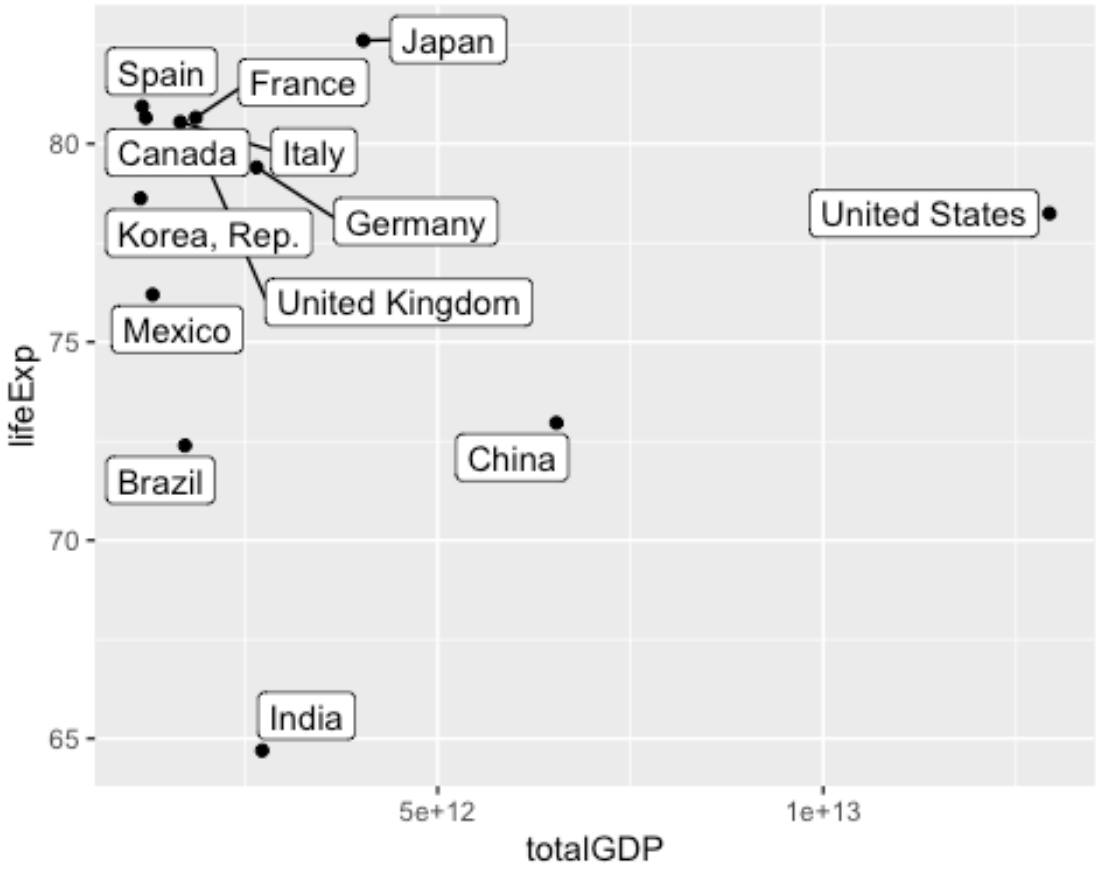
```
dfGapTotGDP %>%
  filter(totalGDP>1e+12) %>%
  ggplot(aes(x= totalGDP, y = lifeExp)) + geom_point() +
  geom_label(aes(x = totalGDP, y = lifeExp, label=country))
```



India, United States, China, Japan, Brazil, Mexico

iv. TTo overcome the poor visibility in the earlier graphs, Install and load the ggrepel library. After that, copy the same code and use geom_label_repel() function instead of geom_label(). Does it look better now? Describe what has changed.

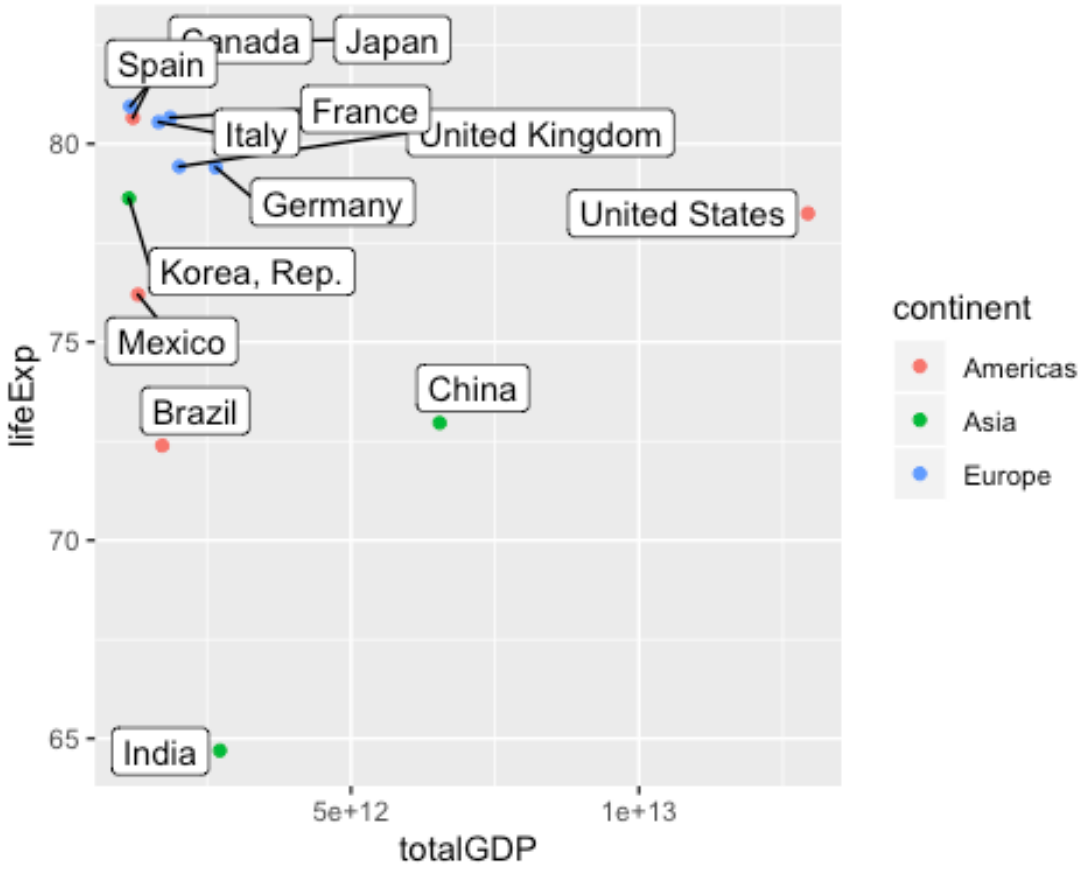
```
library("ggrepel")

dfGapTotGDP %>%
  filter(totalGDP>1e+12) %>%
  ggplot(aes(x= totalGDP, y = lifeExp)) +
  geom_point() +
  geom_label_repel(aes(x = totalGDP, y = lifeExp, label=country))
```



Its look better now, The label is pointed at the particular point without any overlap.

v.Copy the same code. This time, add a color for the continent. What are the continents that are missing from your visual? Why do you think so?

```
library("ggrepel")

dfGapTotGDP %>%
  filter(totalGDP>1e+12) %>%
  ggplot() +
  geom_point(mapping = aes(x= totalGDP, y = lifeExp, color = continent) ) +
  geom_label_repel(aes(x = totalGDP, y = lifeExp, label=country))
```

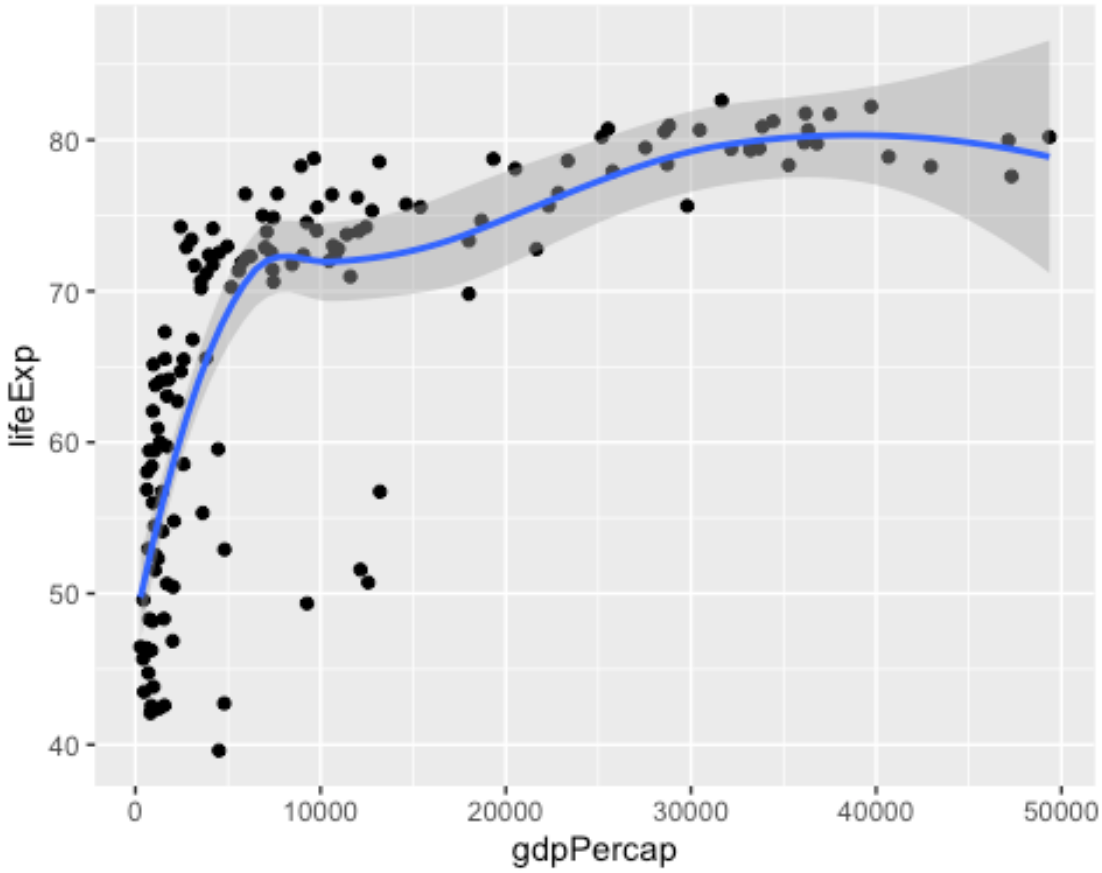


Missing Continents are: Oceania and Africa This continents have countries with total GDP less than 1 trillion $

4.b.You have an idea about the relationship between life expectancy and totalGDP even though you have not tested it statistically. Now, let's examine a more realistic relationship between life expectancy and gdpPercap (GDP per capita). Plot life expectancy (y-axis) against gdpPercap (x-axis) for 2007, add a smoothed line (no need to define any parameters, use the defaults). What do you observe about the overall relationship? Don't use any labels, just focus on the aggregate.

```
dfGapTotGDP %>%
  ggplot(aes(x=gdpPercap, y=lifeExp))+ geom_point()+ geom_smooth()
```

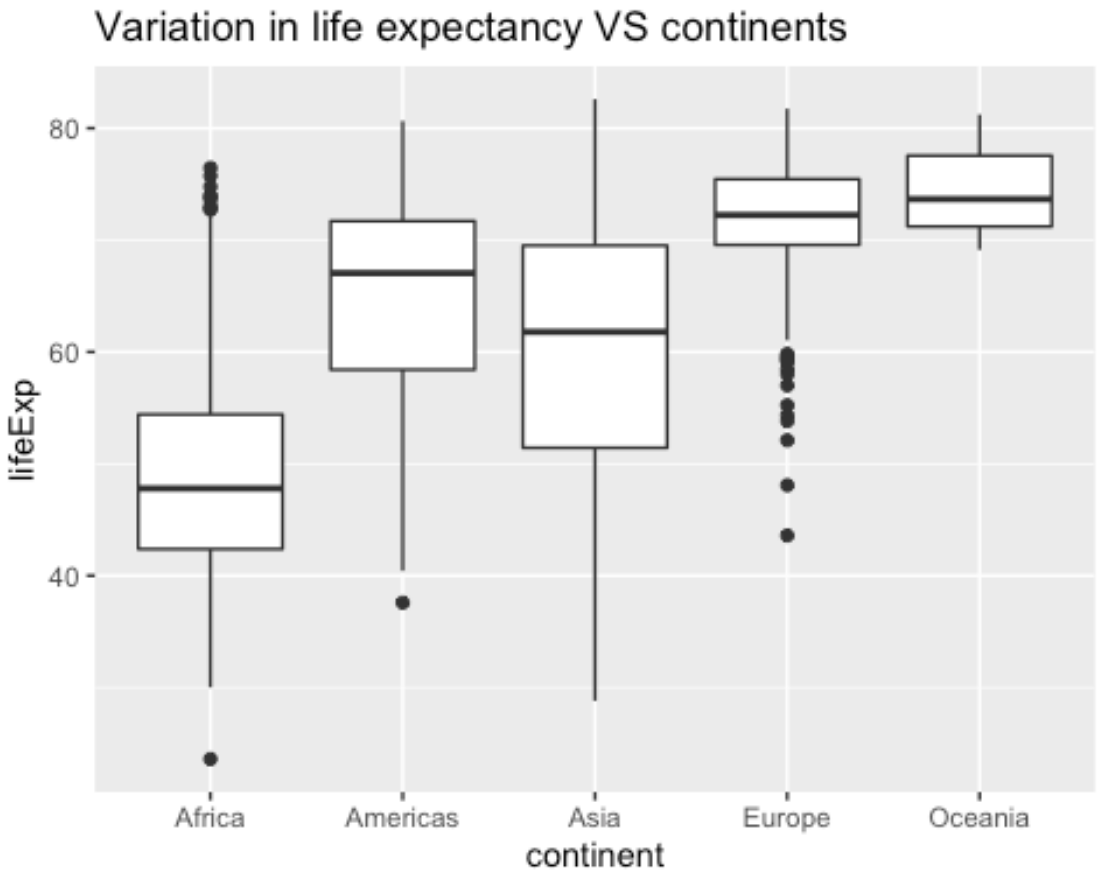
```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



Life expectancy increases drastically for dgpPercap from 0 to 10000, later on growth slows down and became almost stagnant around life expectancy of 80.

4.c.Now let's find out the variations in life expectancy across different continents. Create box plots for each continent (in the same plot) and add a title this time. What do you observe? Describe your observations and answer the questions:

```
dfGap %>%
  ggplot(mapping = aes(x= continent, y = lifeExp))+geom_boxplot()+
  labs(title = "Variation in life expectancy VS continents")
```



i.Which continent has the highest median life expectancy? Oceania has the highest life expectancy

ii.Which continent has the largest range of life expectancy? Asia has the largest range of life expectancy

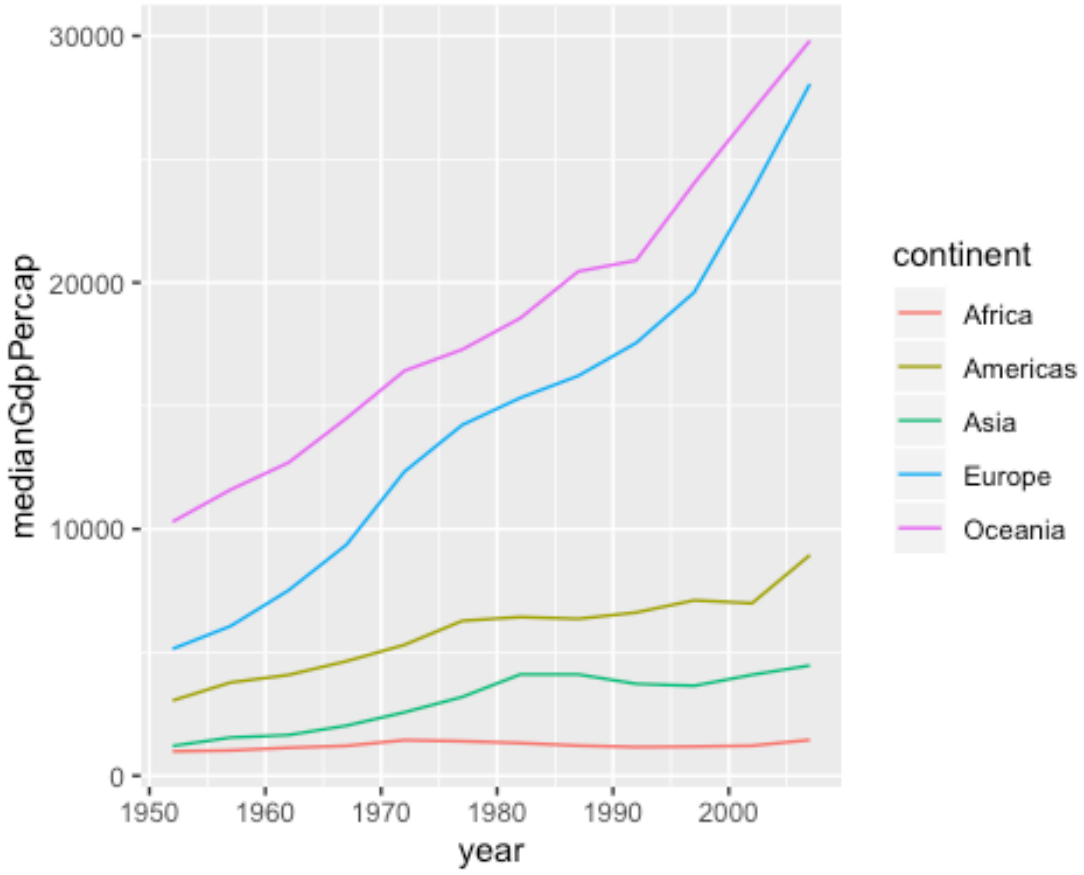
iii.Save your plot as boxPlotsForAll and put it into the ggplotly() function. More useful, right? Report the actual medians per continent by reading from the new interactive plot ggplotly() has created for you.

```
boxPlotsForAll<-
  dfGap %>%
  ggplot(mapping = aes(x= continent, y = lifeExp))+geom_boxplot()+
  labs(title = "Variation in life expectancy VS continents")

ggplotly(boxPlotsForAll)
```

Median life expentency per continent: Africa: 47.79 Americas: 67.05 Asia: 61.79 Europe: 72.24 Oceania: 73.66

4.d.Finally, it is time to create a more advanced (and likely more helpful) plot. Create a line plot to show how median GDP per capita by continent changes over time. [Hint: For the continents, use the color parameter]. Describe what you observe.

```
dfgapyrcontinent <-
  dfGap %>%
  group_by(year, continent) %>%
  summarize(medianGdpPercap = median(gdpPercap))

ggplot(dfgapyrcontinent, aes(x = year, y = medianGdpPercap, color =
continent)) +
  geom_line()
```



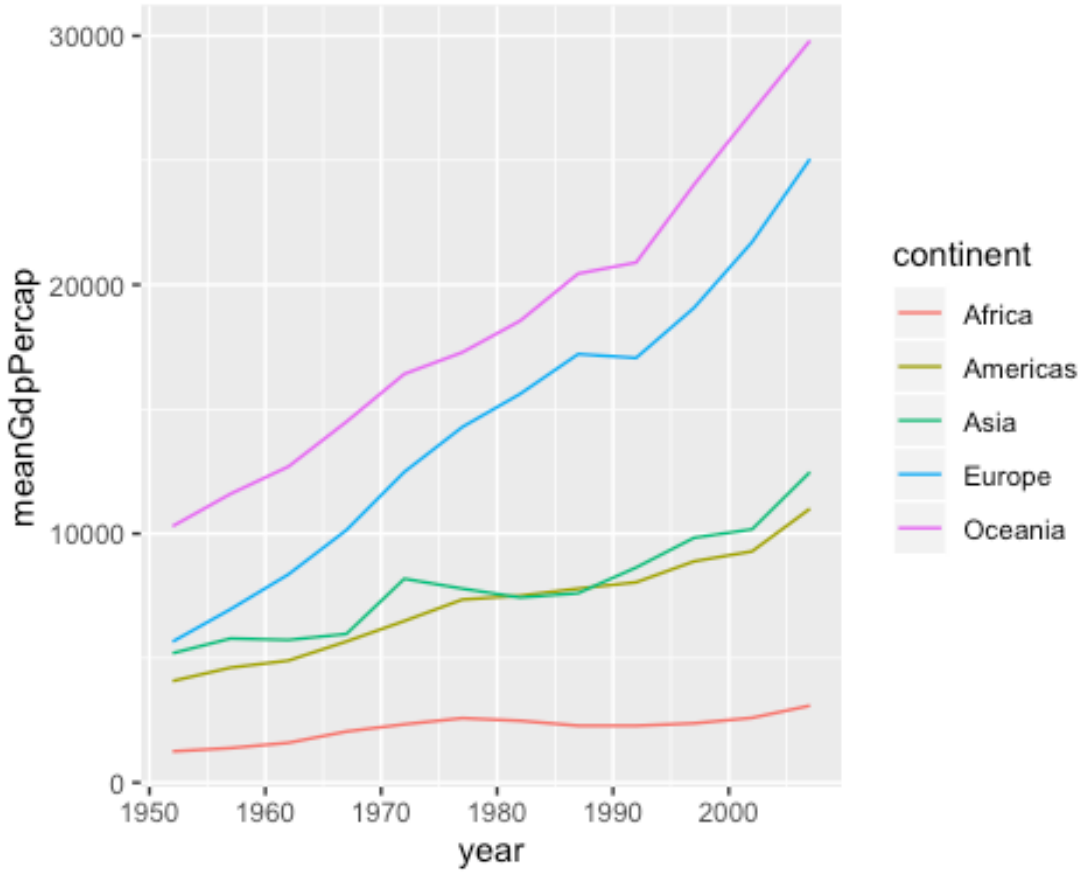
From 1950-2000, median GDP per capita increases slowly for Americas and Asia. We can see fast increasing trend of median GDP per capita for Europe and Oceania whereas for Africa it's almost steady and low.

i.What continents have a clearer trend than others? Why do you think so? Oceania and Europe have clear trend than other because over the years their population has not increased drastically like Asia, Americas and Africa, but their GDP has been boosted. Hence median GDP per cap is for Oceania and Europe shows increasing trend.

ii.Change the summary metric from median to mean. What has changed? Why do you think so?

```
dfgapyrcontinent <-
  gapminder %>%
  group_by(year, continent) %>%
```

```
  summarize(meanGdpPercap = mean(gdpPercap))

ggplot(dfgapyrcontinent, aes(x = year, y = meanGdpPercap, color = continent))
+
  geom_line()
```



After changes, line plot for Asia and Americas almost overlapped. For Asia, Trend is comparatively increasing than previous plot.

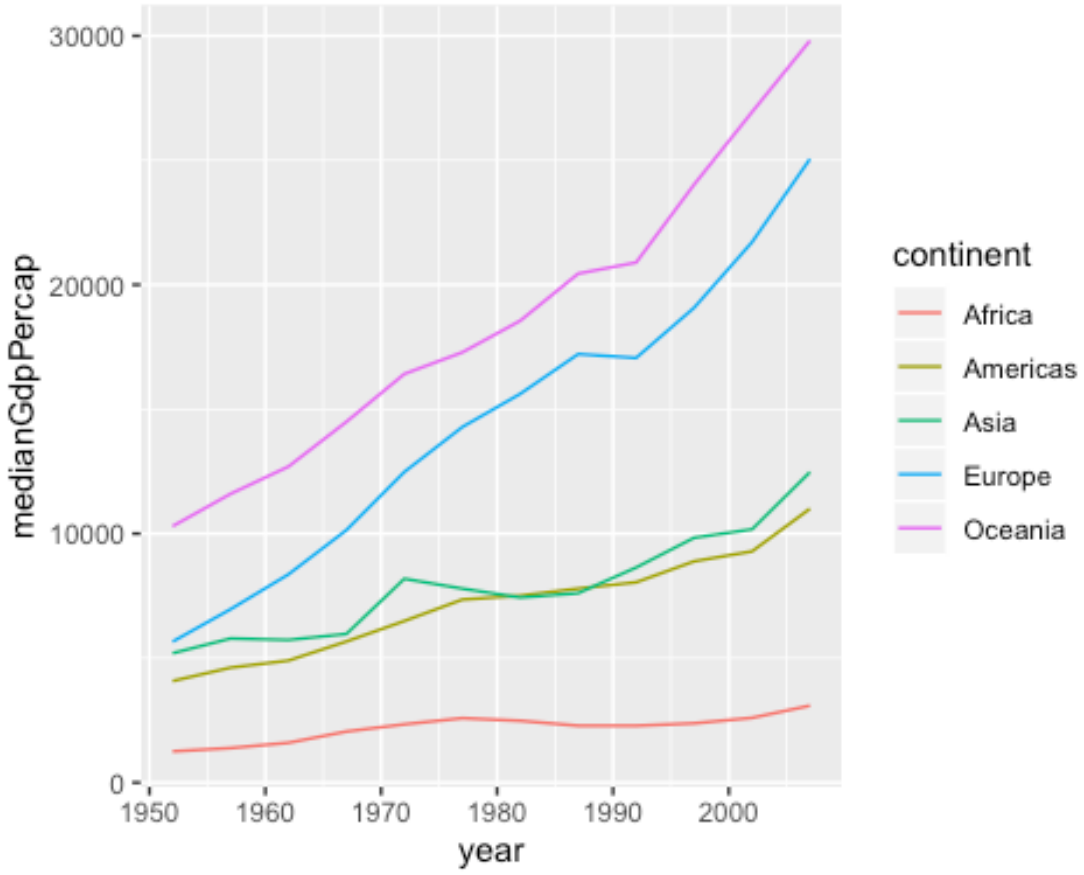
It signifies that huge part of population in Asia has average (Less) GDP & remaining few people has very high GDP. Hence when we plotted Median GDP per Cap, trend was almost constant with insignificant growth.

But, when we plotted mean, we see increasing trend. Whereas for America there are no changes in two plots, average and median GDP per cap is almost same.

iii.Finally, don't you think these plots would be much more useful in plotly? Pick one and save it as gdpOverTime and call ggplotly() on it. You can now read the actual GDP values per year. What are some of the breakthrough years (steep changes) for GDP in different continents?

```
dfgapyrcontinent <-
  gapminder %>%
  group_by(year, continent) %>%
```

```
  summarize(medianGdpPercap = mean(gdpPercap))

gdpOverTime<-ggplot(dfgapyrcontinent, aes(x = year, y = medianGdpPercap,
color = continent)) +
  geom_line()

ggplotly(gdpOverTime)

plot(gdpOverTime)
```



Steep Changes: Oceania: 1992 to 2007 Europe: 1992-2007 Asia: 2002-2007 America: 2002-2007