Explanation

Explanation

There are two major script files were used. The first one is wrangling_code.R, the second one is data_visualization_code.R.

The purpose of wrangling_code.R is to clean and tidy data downloaded from Gapminder and transform it to tidy format. The major packages used in the script is tidyr, tidyverse, dplyr. The logic of this script is first import two csv files downloaded from Gapminder, worthnoting that when import data, fileEnoding command was used, because in the file there are byte order mark (or BOM), which text editor might interpret it incorrectly. Then pivot_longer command is used to reshape both data from wide to long. Next gsub command was used to remove X before year numbers. These operations were performed on both data. The last step is use full_join from dyplr to have full merge, keep observations from both dataframe, therefore, it can be freely filtered later and not lose out any data points when merging. The merged dataframe is agg_df, which includes both indicators, country, and year.

The purpose of data_visualization_code.R is to write a customized function, which takes in data provided and output ideal ggplot. The major package is ggplot2. The plot will be returned from this function is scatterplot.

Then in markdown, there are 3 additional chucnks. The first one is data_clean chucnk. This chunck further clean data. In income var, there are observations that have unit as "k" to represent 1000. To remove it, "k" need to be identified then multiple the number by 1000. The command used is parse_number, which need to be used with charaters. Therefore, the income var was converted to charaters first, and CO2 emissions was converted to numeric.

The second and third chucks are plot chucnks, for 2 subgroups. The first subgroup is for Germany, the second subgroup is China. Both subgroup's time period is filter out to be between 1950 and 2017. Graph title, x-axis title, and y-axis title are also added. The procedure and codes for the two graphs are the same logic.

```
source("wrangling_code.R", echo = T)
```

```
> options(scipen = 6, digits = 4)

> memory.limit(30000000)
[1] Inf

> if (!require("pacman")) install.packages("pacman")

Loading required package: pacman

> pacman::p_load("tidyr", "tidyverse", "dplyr", "data.table",
+     "zoo", "rstudioapi")

> income <- as_tibble(read.csv("income_per_person_gdppercapita_ppp_inflation_adjusted.csv",
+     fileEncoding = "UTF-8-BOM"))

> co2 <- as_tibble(read.csv("co2_emissions_tonnes_per_person.csv",
+     fileEncoding = "UTF-8-BOM"))

> income_long <- income %>% mutate_if(is.numeric, as.character,
```

```
+     is.factor, as.character) %>% pivot_longer(X1799:X2049, names_to = "year",
+    .... [TRUNCATED]

> co2_long <- co2 %>% mutate_if(is.numeric, as.character,
+     is.factor, as.character) %>% pivot_longer(-country, names_to = "year",
+     values_ .... [TRUNCATED]

> agg_df = full_join(x = income_long, y = co2_long,
+     by = c("country", "year"))
```

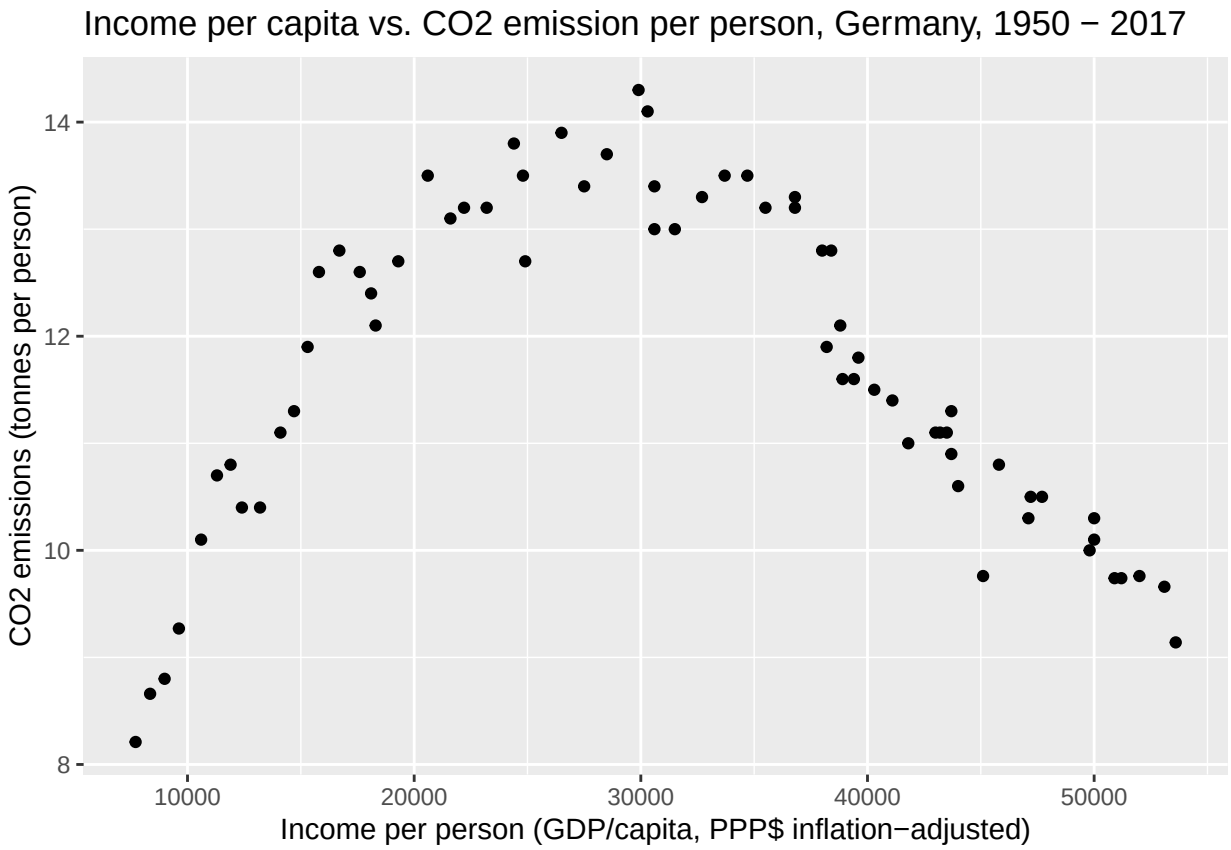
```
source("visualization_function.R", echo = T, prompt.echo = "", spaced = F)
```

```
options(scipen = 6, digits = 4)
memory.limit(30000000)
[1] Inf
if (!require("pacman")) install.packages("pacman")
pacman::p_load("tidyr", "tidyverse", "dplyr", "ggplot2")
ggfun <- function(dat, x.var, y.var) {
+     p <- ggplot(data = dat, aes(x = x.var, y = y.var)) + geom_point()
+     return(p)
+ }
```

```
df_clean <- agg_df %>%
  mutate(income_capita = as.character(income_capita)) %>%
  mutate(co2_capita = as.numeric(co2_capita)) %>%
  mutate(income_capita_k <- case_when(str_detect(income_capita, 'k') ~ parse_number(income_capita)*1e3,
                                      TRUE ~ parse_number(income_capita))) %>%
  select(., c(1,2,4,5)) %>%
  rename(., income_capita = 4)
```

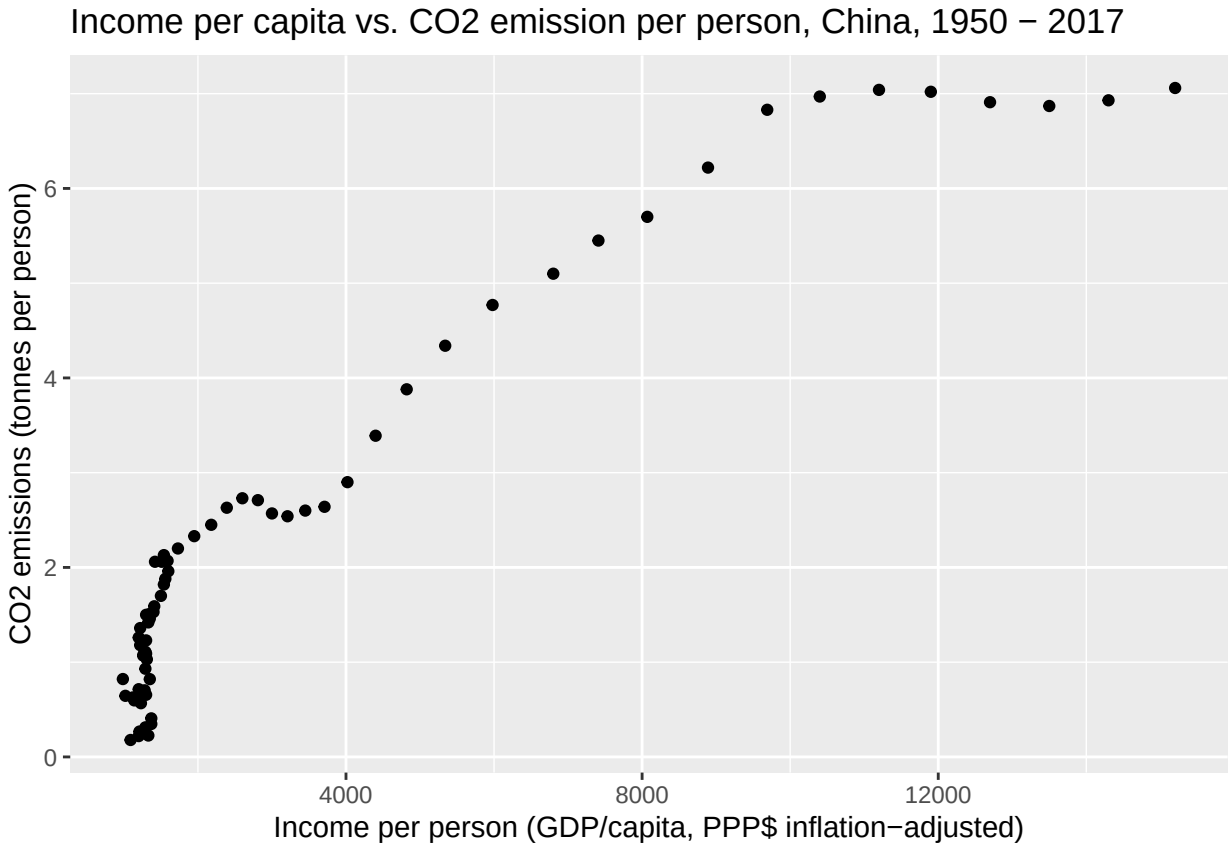
```
df_clean_1 <- df_clean %>%
  filter(., year >= 1950 & year <= 2017 & country == "Germany")

ggfun(df_clean_1, df_clean_1$income_capita, df_clean_1$co2_capita)+
  ggtitle("Income per capita vs. CO2 emission per person, Germany, 1950 - 2017")+
  xlab('Income per person (GDP/capita, PPP$ inflation-adjusted)')+
  ylab("CO2 emissions (tonnes per person)")
```




```
df_clean_2 <- df_clean %>%
  filter(., year >= 1950 & year <= 2017 & country == "China")

ggfun(df_clean_2, df_clean_2$income_capita, df_clean_2$co2_capita)+
  ggtitle("Income per capita vs. CO2 emission per person, China, 1950 - 2017")+
  xlab('Income per person (GDP/capita, PPP$ inflation-adjusted)')+
  ylab("CO2 emissions (tonnes per person)")
```


Income per capita vs. CO2 emission per person, China, 1950 − 2017
CO2 emissions (tonnes per person)
0
2
4
6
4000
8000
12000
Income per person (GDP/capita, PPP$ inflation−adjusted)