Assignment 3- Group 4

Assignment 3

Rachel Ludgero Mia Aydin Loubna Othmani

Problem 1

```
load("Assignment_3.rda")
library(ggplot2)
```

```
Warning: package 'ggplot2' was built under R version 4.3.3
```

```
library(gganimate)
```

```
Warning: package 'gganimate' was built under R version 4.3.3
```

```
library(dplyr)
```

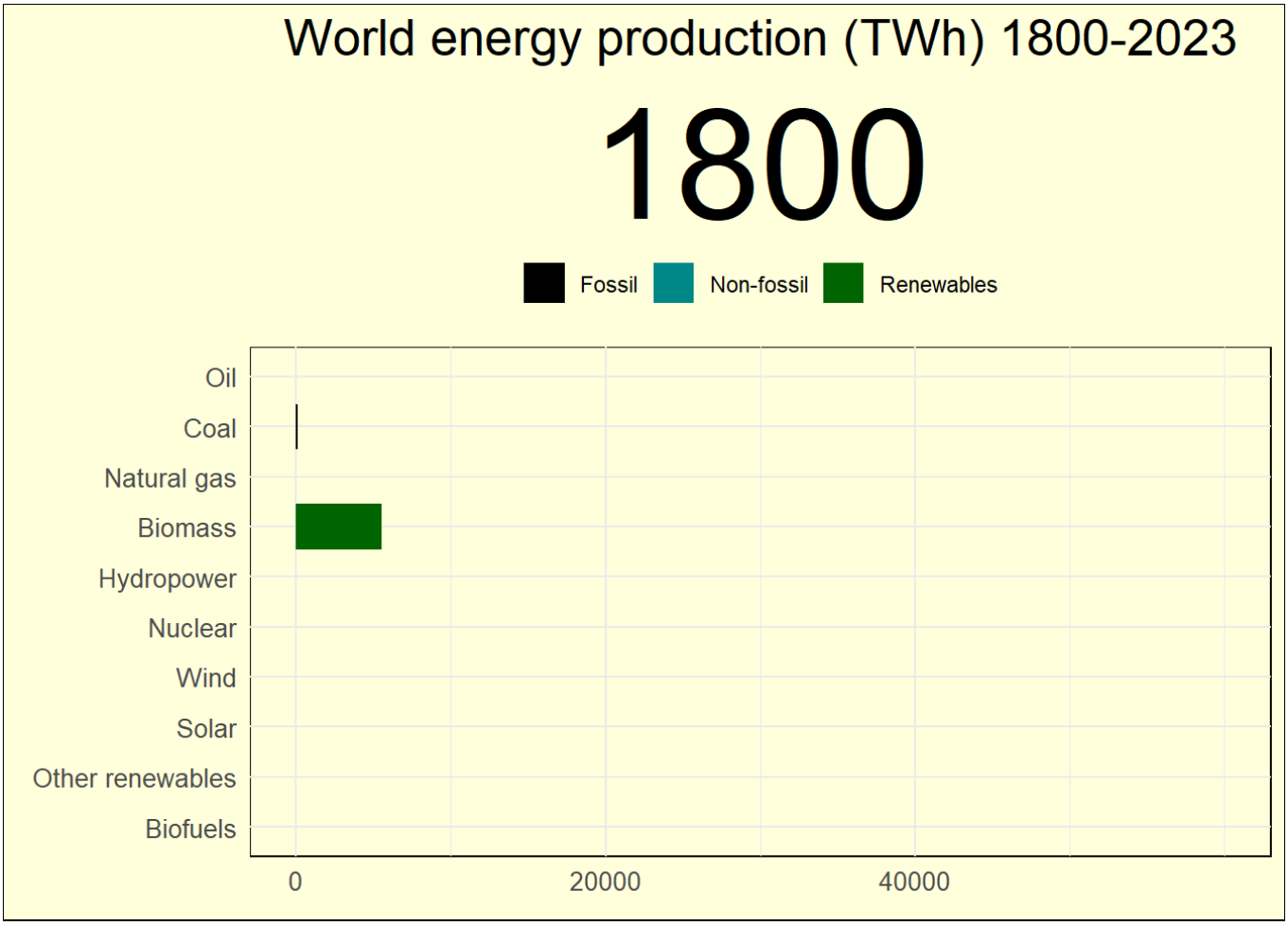
```
Warning: package 'dplyr' was built under R version 4.3.3


Attaching package: 'dplyr'

The following objects are masked from 'package:stats':

    filter, lag

The following objects are masked from 'package:base':

    intersect, setdiff, setequal, union
```

```
library(showtext)
```

```
Warning: package 'showtext' was built under R version 4.3.3

Loading required package: sysfonts

Warning: package 'sysfonts' was built under R version 4.3.3

Loading required package: showtextdb

Warning: package 'showtextdb' was built under R version 4.3.3
```

```
library(gganimate)
animation <- Energy_production %>%
  ggplot(aes(x = Quantity, y = Source, fill = Type)) +
  geom_col() +
  theme_minimal() +
  scale_fill_manual(values = c("black", "darkcyan", "darkgreen")) +
  scale_x_continuous(
    breaks = c(0, 20000, 40000),
```

```
    limits = c(0, 60000)
  ) +
  theme(
    axis.title = element_text(size = 10),
    axis.text.x = element_text(size = 10),
    axis.text.y = element_text(size = 10),
    legend.position = "top",
    panel.background = element_rect(fill = "lightyellow"),
    plot.background = element_rect(fill = "lightyellow"),
    plot.title = element_text(size = 20, hjust = 0.5, face = "bold"),
    plot.subtitle = element_text(size = 60, hjust = 0.5, vjust = 0.5, face = "bold")
  ) +
  labs(
    title = "World energy production (TWh) 1800-2023",
    subtitle = '{frame_time}',
    x = NULL,
    y = NULL,
    fill = NULL
  ) +
  transition_time(Year) +
  ease_aes("linear")

animate(animation,
        nframes = 100,
        fps = 10,
        renderer = gifski_renderer())
```



Problem 2

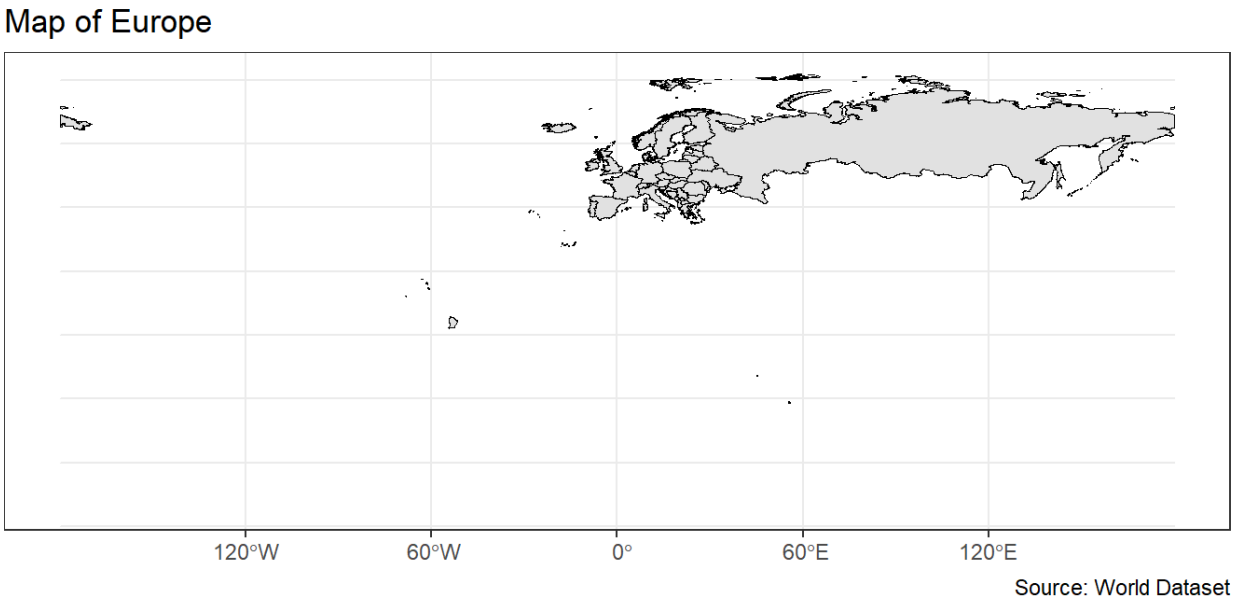
a) Create a new dataset Europe that contains only European countries (you can use the filter function in dplyr on the variable CONTINENT in World).

```
library(sf, quietly = TRUE)
```

```
Warning: package 'sf' was built under R version 4.3.3
```

```
Linking to GEOS 3.11.2, GDAL 3.8.2, PROJ 9.3.1; sf_use_s2() is TRUE
```

```
Europe <-World %>%
  filter(CONTINENT == "Europe")
ggplot(Europe) +
geom_sf(aes(fill = NULL), color ="black") +
theme_bw() +
  labs(title = "Map of Europe",
       caption = "Source: World Dataset")
```



b. Join Europe with some dataset that contains recent information on the European countries. Find a suitable dataset yourselves. A few places to look are The World Bank and Our World in Data.

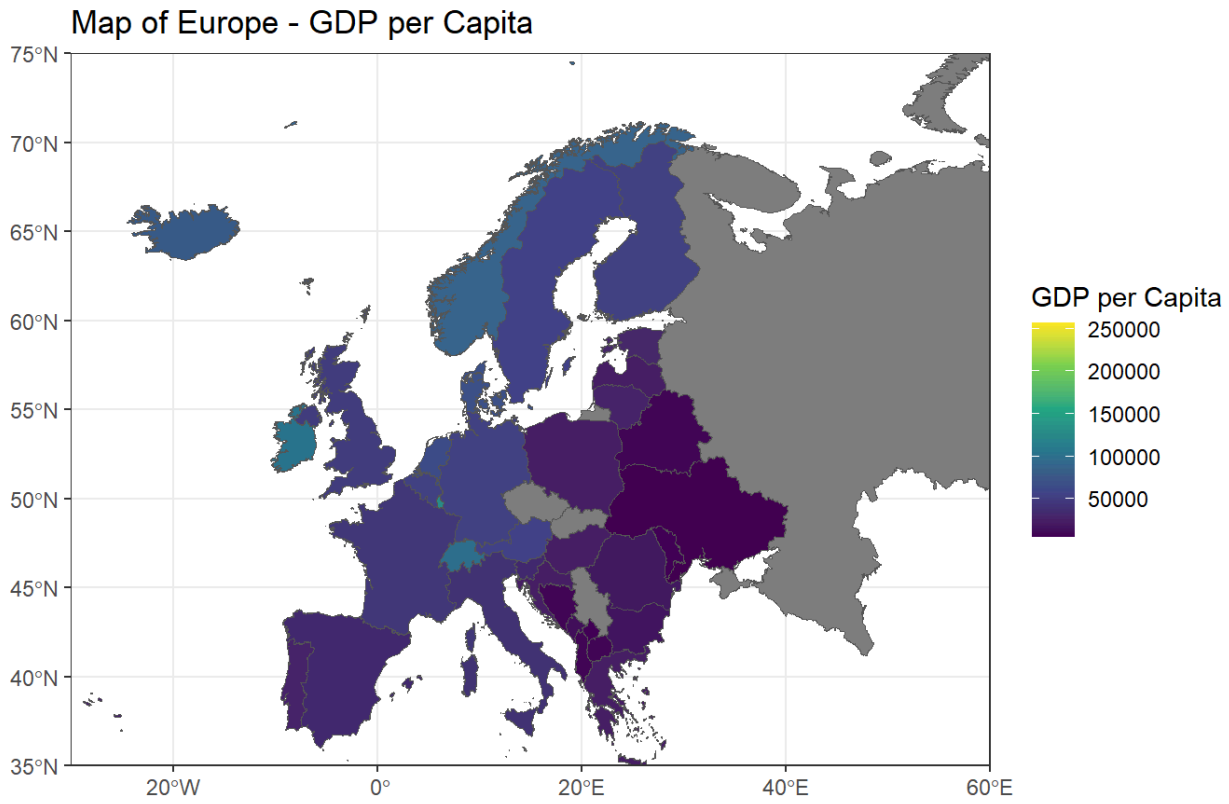
```
library(sf)
library(dplyr)
library(ggplot2)
library(readxl)
```

```
Warning: package 'readxl' was built under R version 4.3.3
```

```
WorldBank <- read_excel("World Bank.xlsx")
WorldBank<- WorldBank %>% distinct(`Country Name`, .keep_all = TRUE)

Europe_Map <- left_join(Europe, WorldBank, by = c("SOVEREIGNT" = "Country Name"))

Europe_Map$`2023` <- as.numeric(Europe_Map$`2023`)
```

```
Warning: NAs introduced by coercion
```

```
ggplot(Europe_Map, aes(fill = `2023`)) +
  geom_sf() +
  theme_bw() +
  labs(title = "Map of Europe - GDP per Capita", fill = "GDP per Capita") +
  coord_sf(xlim = c(-30, 60), ylim = c(35, 75), expand = FALSE) +
  scale_fill_viridis_c(option = "viridis")
```



c) Create an interactive map of Europe using ggiraph or ggplotly. Use the fill color to display some information about the countries and also add interactive functionality so that a tooltip with some pieces of information on a country is displayed when you hover over it with the mouse pointer. You might want to crop in a bit with xlim and ylim (you don't have to include all of Russia and the Canary islands.

```
library(sf)
library(dplyr)
library(ggplot2)
library(readxl)
library(ggiraph)
```

```
Warning: package 'ggiraph' was built under R version 4.3.3
```

```
library(GiRaF)
```

```
Warning: package 'GiRaF' was built under R version 4.3.3
```

```
WorldBank <- WorldBank %>% distinct(`Country Name`, .keep_all = TRUE)



Europe_Map <- left_join(Europe, WorldBank, by = c("SOVEREIGNT" = "Country Name"))


Europe_Map$`2023` <- as.numeric(Europe_Map$`2023`)
```

```
Warning: NAs introduced by coercion
```

```
gg_map <- ggplot(Europe_Map, aes(fill = `2023`, tooltip = paste("Country: ", SOVEREIGNT, "<br
  geom_sf_interactive() +
  theme_bw() +
  labs(title = "Interactive Map of Europe - GDP per Capita", fill = "GDP per Capita") +
  coord_sf(xlim = c(-30, 60), ylim = c(35, 75), expand = FALSE) +
  scale_fill_viridis_c(option = "viridis")

girafe(ggobj = gg_map)
```