The Proper Set-Up

First, set your working directory using the setwd() function. Then, load in the needed libraries with the library() function.

```r
# Each person's setwd() function will look a little different, 
# depending on where they have saved the training data folder. 
# It may look like: 
# setwd("/Volumes/GoogleDrive/My Drive/Training Data August 2021")
library(ggplot2) # ggplot2 creates graphs
library(tidyverse) # tidyverse is used for data management
library(readr) # readr is needed for the read_csv() function
```

For this exercise, you need to load in `wdi_cleaned_part2.csv`.

```r
dat <- read_csv("wdi_cleaned_part2.csv")
```

Following these steps, we can start re-creating the two graphs!

Pay attention to details such as:
1) the scaling and labeling of the axes
2) the line types
3) the background of the plot
4) the information included, such as the title and data source
5) the legends

Try to match the colors as closely as possible, unless you can do it better (“challenge accepted”). Some skills required to re-create the precise details of these graphs were not covered in the walkthrough work. However, troubleshooting and googling are an important part of R. So . . happy hunting!

Re-creating Graph 1

```r
# remember the rough structure for coding a graph is the following: 
ggplot(data = , aes(x = , y = , color = , linetype = )) + geom() + 
#[other graphical parameters, e.g. title, color schemes, background]
```
ggplot(subset(dat, year %in% c(1992, 2002, 2012)),
# we use the subset command within the ggplot command to select the years

aes(x = renewable_energyuse,
    color = factor(year),
    linetype = factor(year))) +

# here we assign renewable_energyuse variable to the x axis.
# then, we use the color and line parameter to make each color and linetype
# subject to a different year

geom_line(stat = "density") +

# the geom_line() geometric object function makes it a line graph. since we do
# not want the density graph to be plotted as a closed polygon, we add the
# stat = "density" parameter.

labs(title = "Distribution of renewable energy use across all countries",
     subtitle = "Data source: World Development Indicators",
     x = "Renewable energy consumption (% of total final energy consumption)",
     y = "Density") +

# here we are adding a title and subtitle as well as labels for the x and y axis.

theme_bw() +

# this gives the graph a black and white theme.

scale_color_manual(name = "Year",
    values = c("darkorange",
                "cyan",
                "blue")) +

# we use scale_color_manual() to create a discrete scale based on color.
# first, we add the name parameter, "Year", and then we assign color values.
# because we do not specifically which specific year is assigned to which
# specific color, the values will be matched numerical order
# (first color will be matched to the first number value given).

scale_linetype_manual(name = "Year",
    values = c("dotted",
                "solid",
                "dashed")) +

# we use scale_linetype_manual() to create a discrete scale based on line type.
# first, we add the name parameter, "Year", and then assign
# different linetype values.

theme(legend.key.size = unit(1, "cm")) +

# here we changed the size of the legend key.
Below, the `coord_cartesian()` function zooms the plot while not modifying the underlying data. by including the `ylim` parameter we set limits on the y axis. More specifically, we set the minimum at 0 and the maximum at 0.02

```r
coord_cartesian(ylim = c(0, 0.02))
```

Distribution of renewable energy use across all countries

Data source: World Development Indicators

Re-creating Graph 2

```r
ggplot(subset(dat, country %in% c("United States", "Germany", "Brazil", "Russian Federation", "India", "China")),
#we use the subset command within the ggplot command to select the countries "United States, Germany, Brazil, Russian Federation, India, China from the "country" column.

aes(x = energyuse_pop)) +
#here we have assigned energyuse_pop variable to the x axis.

geom_density(fill = "blue", alpha = 0.3, color = NA) +
#first we use the geom_density() function to create a density plot
#of the energyuse_pop variable. the alpha parameter adjusts the
#opacity of the line. NA being assigned to the color parameter prevents any
#color.

```r
facet_wrap(~ factor(country), ncol = 2) +
```

#we use the facet_wrap() function to draw a separate plot for each country.
#it will only plot the countries we subsetted earlier in the code.
#then we set the ncol parameter to 2 to determine the number of columns.

```r
theme_bw() +
```

#this gives the graph a black and white theme.
#Below we are adding a title and subtitle as well as labels for the x and y axis.

```r
labs(title = "Per capita energy consumption between 1992 and 2014",
     subtitle = "Data source: World Development Indicators",
     x = "Energy use (kg of oil equivalent per capita)",
     y = "Density")
```
Per capita energy consumption between 1992 and 2014
Data source: World Development Indicators

<table>
<thead>
<tr>
<th>Country</th>
<th>Density</th>
</tr>
</thead>
<tbody>
<tr>
<td>Brazil</td>
<td>0.006</td>
</tr>
<tr>
<td>China</td>
<td>0.000</td>
</tr>
<tr>
<td>Germany</td>
<td>0.006</td>
</tr>
<tr>
<td>India</td>
<td>0.000</td>
</tr>
<tr>
<td>Russian Federation</td>
<td>0.006</td>
</tr>
<tr>
<td>United States</td>
<td>0.000</td>
</tr>
</tbody>
</table>

Energy use (kg of oil equivalent per capita)