Introduction to `ggplot2` - Solutions

Exercise A - (2 min)

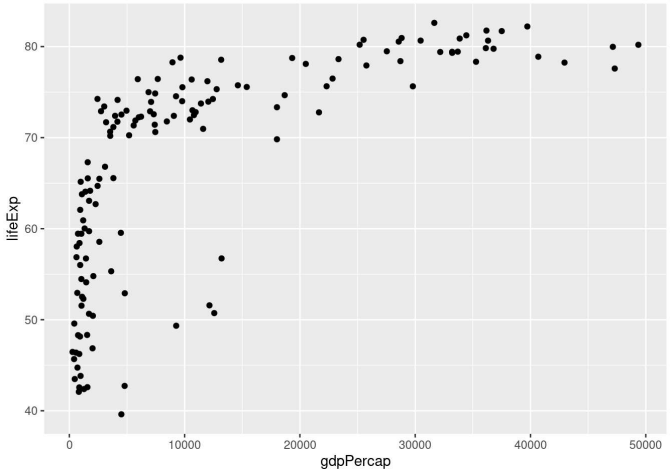
1. Using the preceding slide as a template, make a scatterplot with `pop` on the x-axis and `lifeExp` on the y-axis, based on `gapminder_2007`.
2. Repeat the preceding but with `gdpPercap` on the y-axis.

Solution

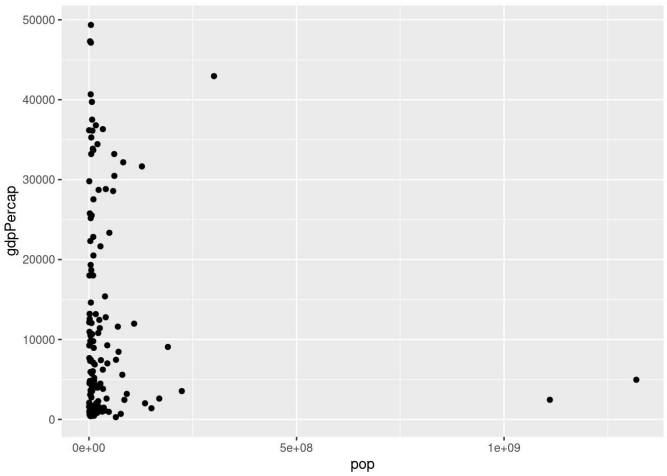
```
# Prep
library(gapminder)
library(tidyverse)

gapminder_2007 <- gapminder %>%
  filter(year == 2007)

# Part 1
gapminder_2007 |>
ggplot(aes(x = gdpPercap, y = lifeExp)) +
  geom_point()
```




```
# Part 2
gapminder_2007 |>
ggplot(aes(x = pop, y = gdpPercap)) +
  geom_point()
```



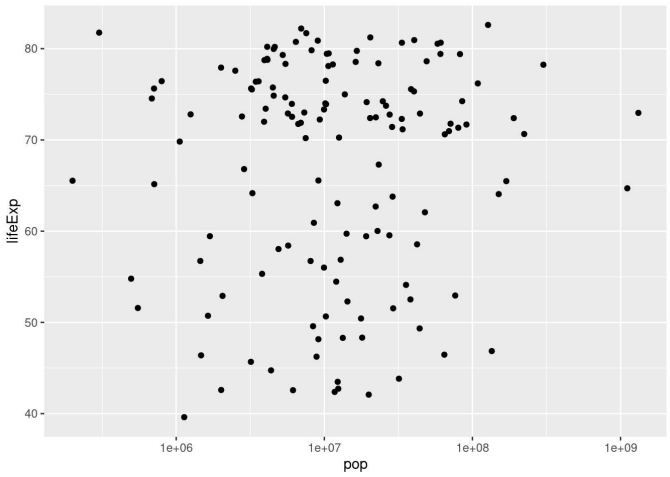
Exercise B - (5 min)

Label your axes and give each plot a title!

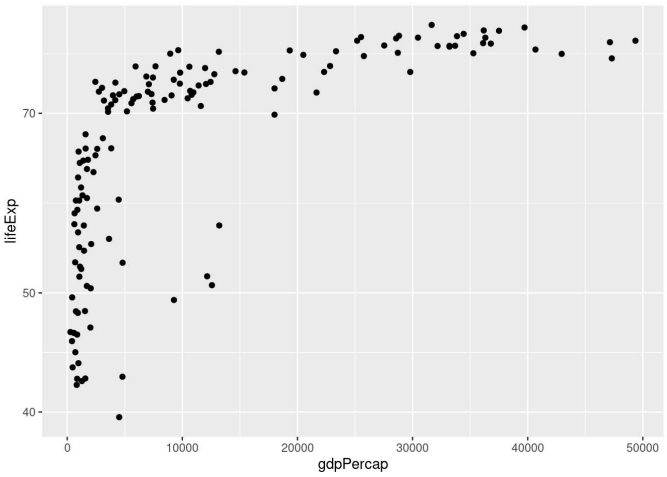
1. Make a scatterplot with the log base 10 of `pop` on the x-axis and `lifeExp` on the y-axis using `gapminder_2007`.
2. Figure out how to make a plot with the *y-axis* on the log scale. Then repeat my plot from the previous slide with `gdpPercap` in levels and `lifeExp` in logs.
3. Repeat 2 but with *both* axes on the log scale.

Solution

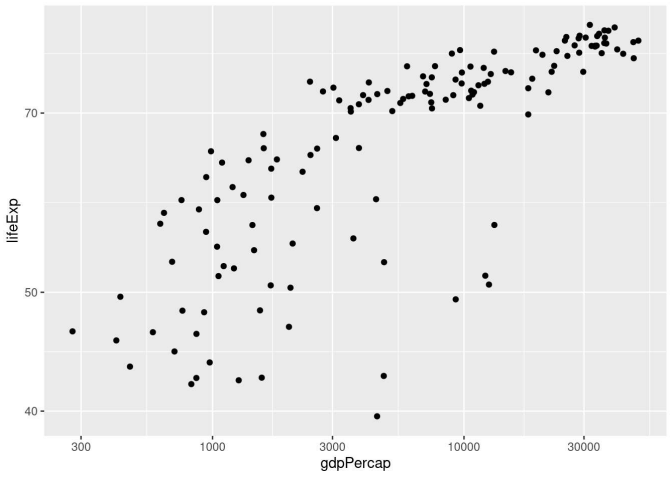
```
# Part 1
gapminder_2007 |>
  ggplot(aes(x = pop, y = lifeExp)) +
  geom_point() +
  scale_x_log10()
```

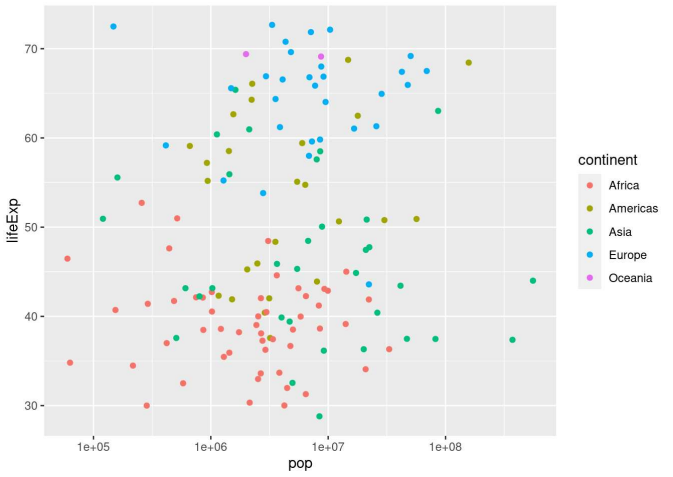



```
# Part 2
gapminder_2007 |>
  ggplot(aes(x = gdpPercap, y = lifeExp)) +
  geom_point() +
  scale_y_log10()
```




```
# Part 3
gapminder_2007 |>
  ggplot(aes(x = gdpPercap, y = lifeExp)) +
  geom_point() +
  scale_x_log10() +
  scale_y_log10()
```





Exercise C - (2 min)

1. Would it make sense to set `size = continent`? What about setting `col = pop`?
2. Using `gapminder` data from 1952, plot life expectancy on the y-axis and log population on the x-axis. Color the points by continent.

Solution

1. Neither of these makes sense since `continent` is `categorical` and `pop` is continuous: color is useful for categorical variables and size for continuous ones.
2. Run the following:

```
gapminder |>
  filter(year == 1952) |>
  ggplot(aes(x = pop, y = lifeExp, color = continent)) +
  geom_point() +
  scale_x_log10()
```

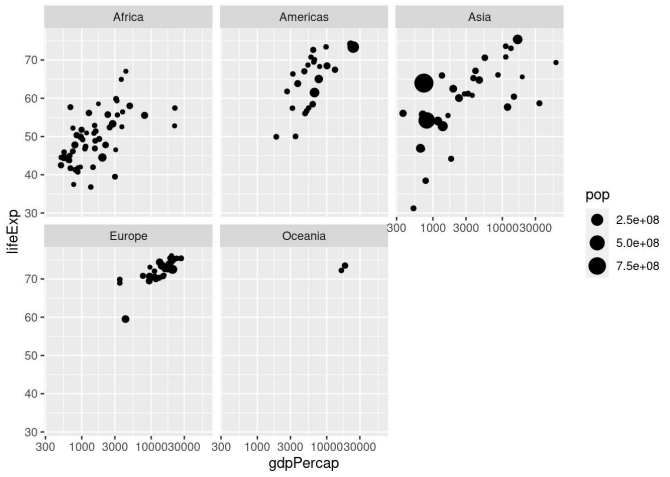
Exercise - D (3 min)

1. Make a scatterplot of `gapminder` data from 1997. Facet by continent and put GDP/capita on the log scale on the x-axis and life expectancy on the y-axis. Indicate population by the size of each point.
2. What do you think would happen if we had tried to facet by `pop` rather than year? Why?

Solution

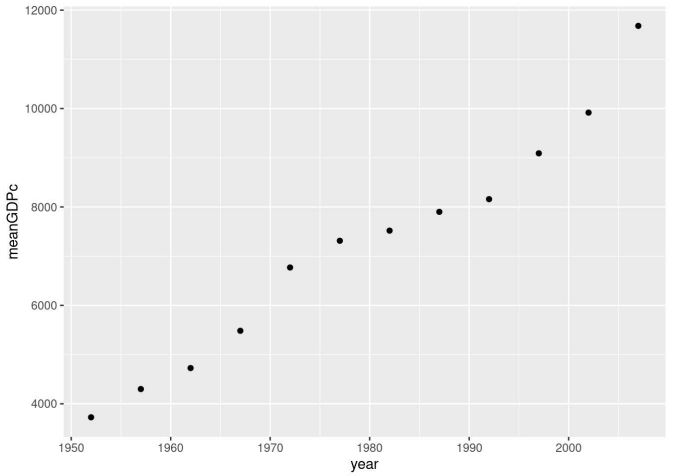
1. Run the following:

```
gapminder %>%
  filter(year == 1977) |>
  ggplot(aes(x = gdpPercap, y = lifeExp, size = pop)) +
  geom_point() +
  scale_x_log10() +
  facet_wrap(~ continent)
```



2. You'll get something crazy if you try this. Population is continuous rather than categorical so every country has a different value for this variable. You'll end up with one plot for every country, containing a single point.

```
  summarize(meanGDPc = mean(gdpPercap)) |>
  ggplot(aes(x = year, y = meanGDPc)) +
  geom_point()
```



3. Run the following:

```
gapminder |>
  group_by(year, continent) |>
  summarize(meanGDPc = mean(gdpPercap)) |>
  ggplot(aes(x = year, y = meanGDPc, color = continent)) +
  geom_point() +
  scale_y_log10()
```

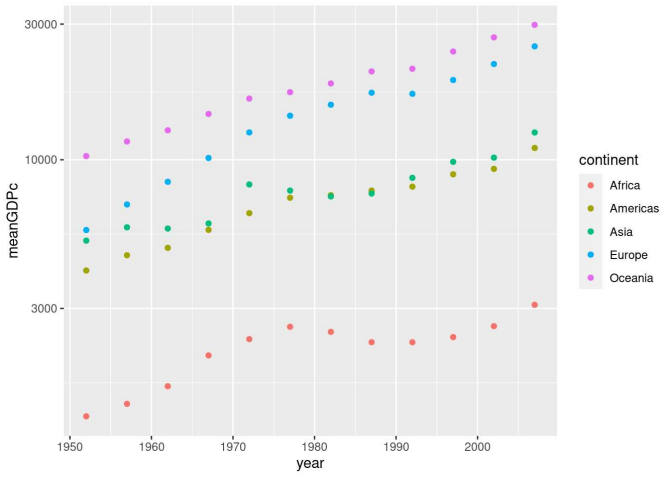
Exercise E - (3 min)

1. Try appending `expand_limits(y = 0)` to the previous plot. What happens? Why and when might this be helpful?
2. Make a scatterplot with average GPD/capita across all countries contained in `gapminder` on the y-axis and `year` on the x-axis.
3. Repeat the preceding, broken down by continent, using color to distinguish the points. Put mean GPD/capita on the log scale.
4. Modify the last plot to include *both* points and lines.

Solution

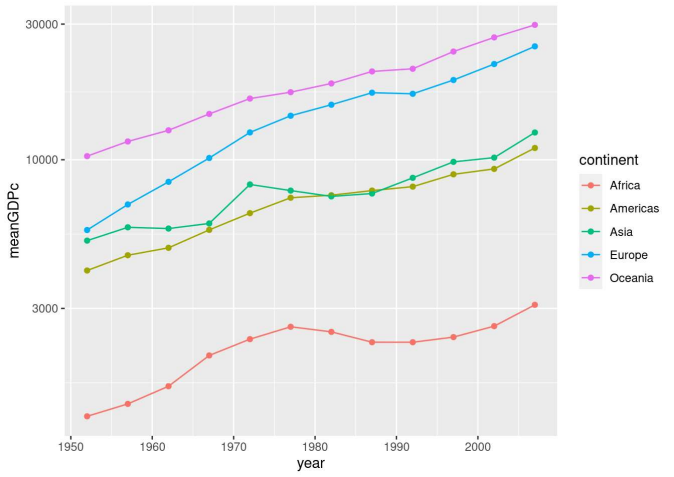
1. The function `expand_limits()` lets us tweak the limits of our x or y-axis in a ggplot. In this particular example `expand_limits(y = 0)` ensures that the y-axis begins at zero. Without using this command, ggplot will choose the y-axis on its own so that there is no "empty space" in the plot. Sometimes we may want to override this behavior.
2. Run the following:

```
gapminder |>
  group_by(year) |>
```



4. Run the following:

```
gapminder |>
  group_by(year, continent) |>
  summarize(meanGDPc = mean(gdpPercap)) |>
  ggplot(aes(x = year, y = meanGDPc, color = continent)) +
  geom_point() +
  geom_line() +
  scale_y_log10()
```



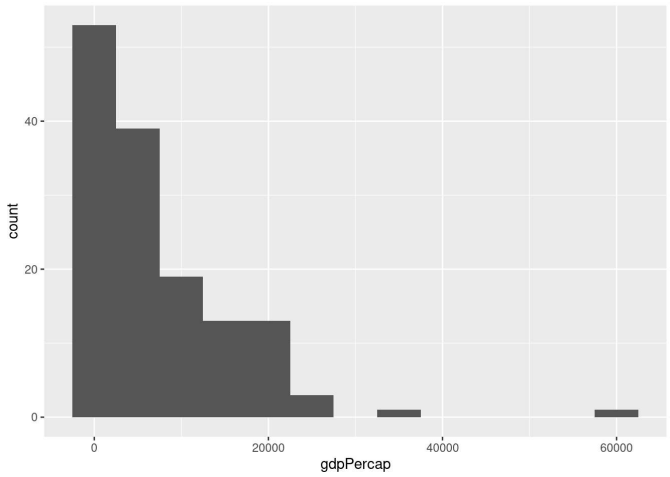
Exercise F - (3 min)

1. What happens if you don't specify a `binwidth`? Try it and find out!
2. Make a histogram of GDP/capita across countries in 1977. Play around with different binwidths until you find one that gives a good summary of the data.
3. Repeat the preceding but with GDP/capita on the log scale. Compare and contrast.

Solution

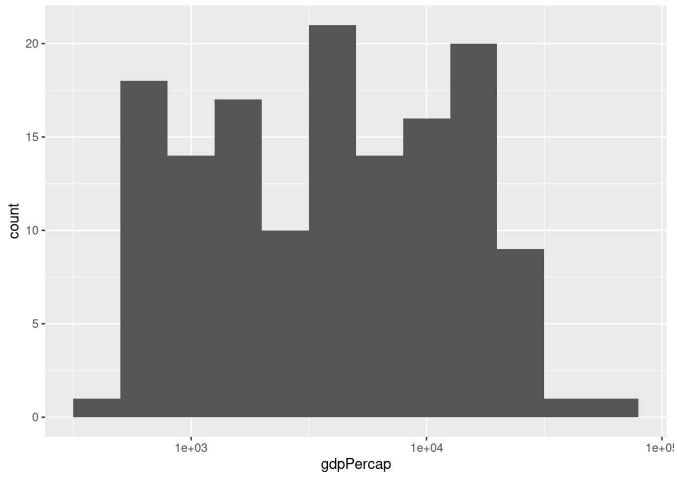
1. If you don't specify a bin width, `ggplot2` will pick one for you and give you a warning suggesting that you pick a better bin width manually.
2. There's no obvious *right answer* for bin width, but here's one possibility:

```
gapminder |>
  filter(year == 1977) |>
  ggplot(aes(x = gdpPercap)) +
  geom_histogram(binwidth = 5000)
```



3. There's no right answer here: it's a discussion question! But a key feature worth noticing is that taking logs mainly eliminates the huge positive skewness in GDP/capita:

```
gapminder |>
  filter(year == 1977) |>
  ggplot(aes(x = gdpPercap)) +
  scale_x_log10() +
  geom_histogram(binwidth = 0.2)
```

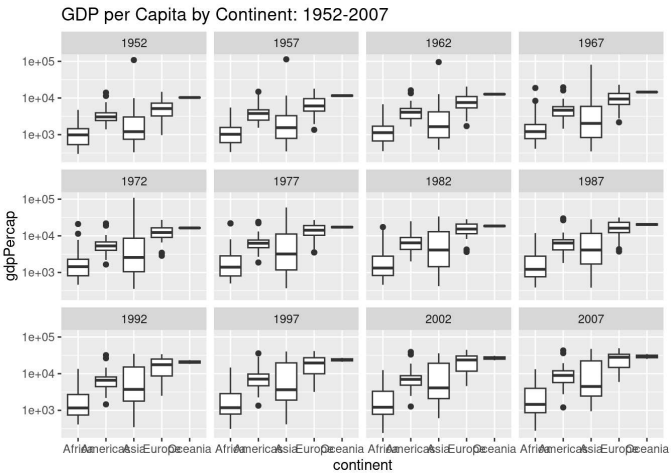


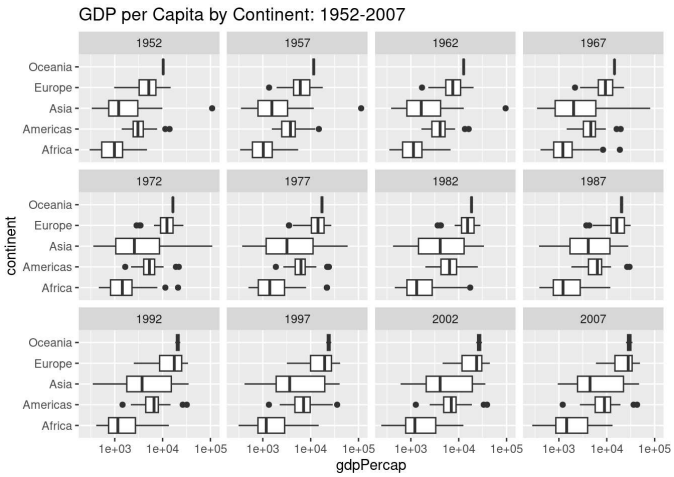
Exercise G - (2 min)

Use faceting to construct a collection of boxplots, each of which compares log GDP/capita across continents in a given year.

Solution

```
gapminder |>
  ggplot(aes(x = continent, y = gdpPercap)) +
  geom_boxplot() +
  facet_wrap(~ year) +
  scale_y_log10() +
  ggtitle('GDP per Capita by Continent: 1952-2007')
```





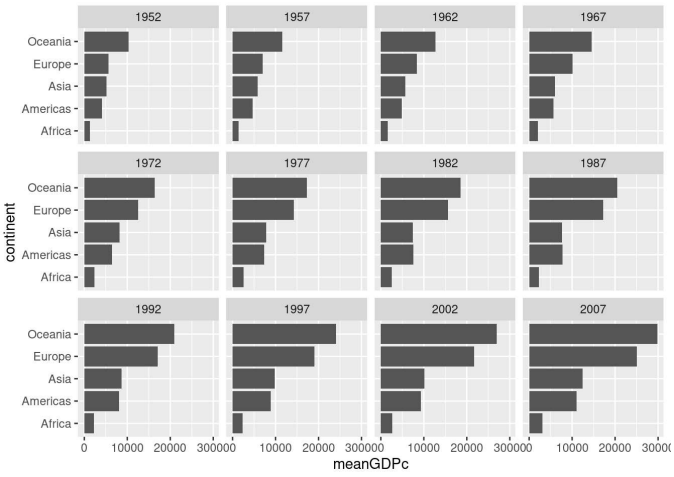
Exercise H - (3 min)

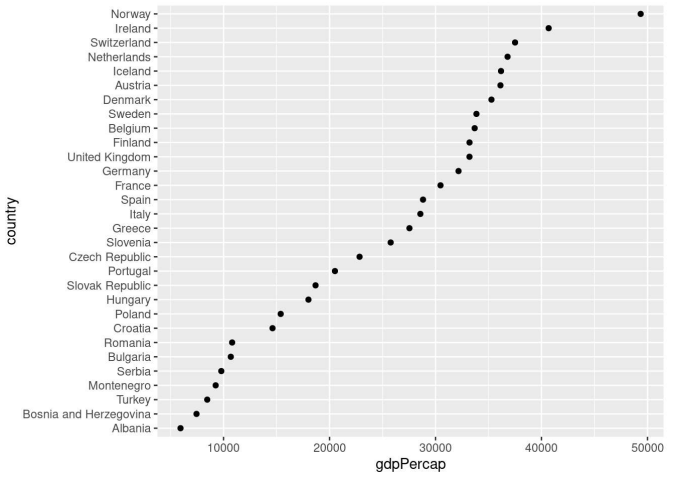
1. Go back and turn your boxplots from the last exercise sideways to make it easier to read the continent labels.
2. Make a collection of bar plots faceted by year that compare mean GDP per capita across countries in a given year. Orient your plots so it's easy to read the continent labels.

Solution

```
# Part 1
gapminder |>
  ggplot(aes(x = continent, y = gdpPercap)) +
  geom_boxplot() +
  facet_wrap(~ year) +
  scale_y_log10() +
  coord_flip() +
  ggtitle('GDP per Capita by Continent: 1952-2007')
```

```
# Part 2
gapminder |>
  group_by(year, continent) |>
  summarize(meanGDPc = mean(gdpPercap)) |>
  ggplot(aes(x = continent, y = meanGDPc)) +
  geom_col() +
  facet_wrap(~ year) +
  coord_flip()
```





Exercise I - (3 min)

Make a dot chart of GDP per capita in all European countries in the year 2007. Sort the dots so that the country with the highest GDP per capita appears a the top and the country with the lowest appears at the bottom.

Solution

```
gapminder %>%
  filter(continent == 'Europe', year == 2007) %>%
  mutate(country = fct_reorder(country, gdpPercap)) %>%
  ggplot(aes(x = gdpPercap, y = country)) +
  geom_point()
```