## **Hands-On Activity: Visualizing data with ggplot2**

**TOTAL POINTS 1**

1.

Question 1



In this activity, you’ll learn the basics of data visualization in ggplot2 through an overview of the basic logic of data visualization in ggplot2. Then, you'll learn how to create a plot in ggplot2 using R code. Finally, we’ll check out the ggplot2 cheat sheet, a useful reference guide for working with ggplot2.

### Grasping the basics of ggplot2



The ggplot2 package lets you make high-quality, customizable plots of your data. As we mentioned in the last video, ggplot2 is based on the grammar of graphics, a system for describing and building data visualizations. The essential idea behind the grammar of graphics is that you can build any plot from the same basic components, or building blocks. These include:

* A dataset.
* A set of geoms. A geom refers to the geometric object used to represent your data. For example, you can use points to create a scatterplot, bars to create a bar chart, lines to create a line diagram, and so on.
* A set of aesthetic attributes. An aesthetic is a visual property of an object in your plot. You can think of an aesthetic as a connection, or mapping, between a visual feature in your plot and a variable in your data. For example, in a scatterplot, aesthetics include things like the size, shape, color, or location (x-axis, y-axis) of your data points.

So, to create a plot with ggplot2, you first choose a dataset to work with. Then, you determine how to visually organize your data on a coordinate system by choosing a geom to represent your data points and aesthetics to map your variables.

### Preparing your data



The ggplot2 package let’s you use R code to specify the dataset, geom, and aesthetics of your plot. Let’s check out an example and learn how to create a plot using code.

First, choose a dataset to work with. In this activity we will make use of the Palmer Penguins data that you’re already familiar with from earlier videos. However, you have the option of using any data set that you wish.

Once you’ve decided on your dataset, open RStudio. Use the install.packages() function to install both ggplot2 and the Palmer Penguins data set.

**install.packages(“ggplot2”)**

**install.packages(“palmerpenguins”)**

Then load ggplot2 and the dataset using the library() function.

**library(ggplot2)**

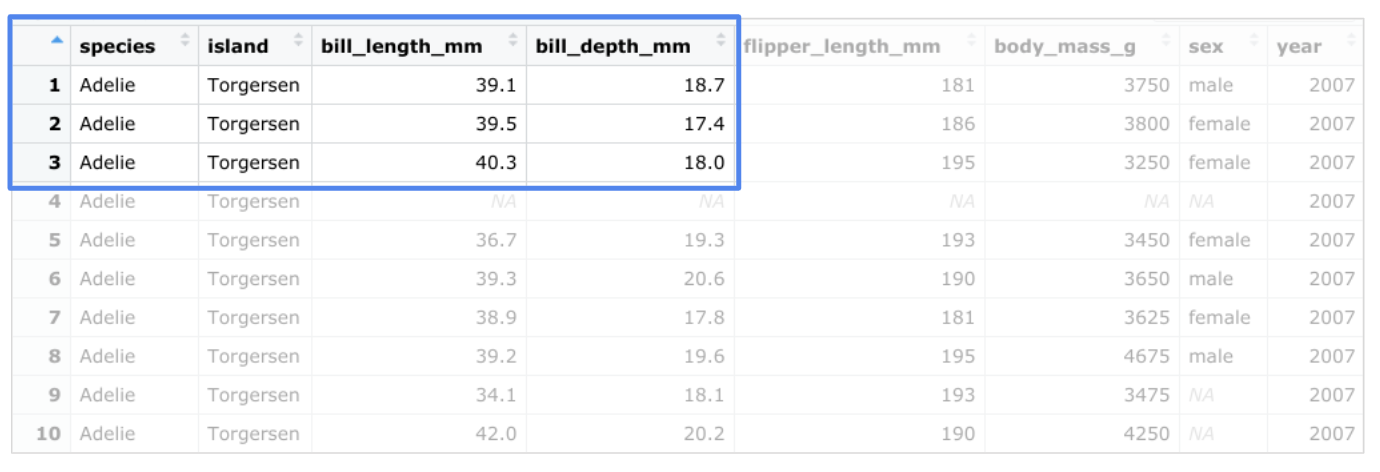
**library(palmerpenguins)**

Let’s take a look inside the data frame for the penguins data. To do this, you’ll use the data() and View() functions. Use a capital “V” for the View() function since functions in R are case sensitive.

**data(penguins)**

**View(penguins)**

This brings up the data frame. Here’s an image of the first ten rows:



The penguins dataset contains size measurements for three penguin species (Adelie, Chinstrap, and Gentoo) that live on the Palmer Archipelago in Antarctica. There are variables such as body mass, flipper length, and bill length.

### Creating a plot in ggplot2



Let’s say we want to plot the relationship between body mass and flipper length in the three penguin species. You can choose a specific geom that fits the type of data you have. Points show the relationship between two quantitative variables. A scatterplot of points would be an effective way to display the relationship between the two variables. We can put flipper length on the x-axis and body mass on the y-axis.

You’ll type the following code to create the plot. But before you run it, let’s review the code piece by piece:

**ggplot(data = penguins) +**

**geom\_point(mapping = aes(x = flipper\_length\_mm, y = body\_mass\_g))**

In ggplot2, you begin a plot with the ggplot() function. The ggplot() function creates a coordinate system that you can add layers to. The first argument of the ggplot() function is the dataset to use in the plot. In this case, it’s “penguins.” In the parentheses, you put “data = penguins”

**ggplot(data = penguins)**

Then, you add a “+” symbol to add a new layer to your plot. You complete your plot by adding one or more layers to ggplot().

**ggplot(data = penguins) +**

Next, you choose a geom by adding a geom function. The geom\_point() function uses points to create scatterplots, the geom\_bar function uses bars to create bar charts, and so on. In this case, we choose the geom\_point function to create a scatter plot of points. The ggplot2 package comes with many different geom functions. You’ll learn more about geoms in upcoming videos.

**ggplot(data = penguins) +**

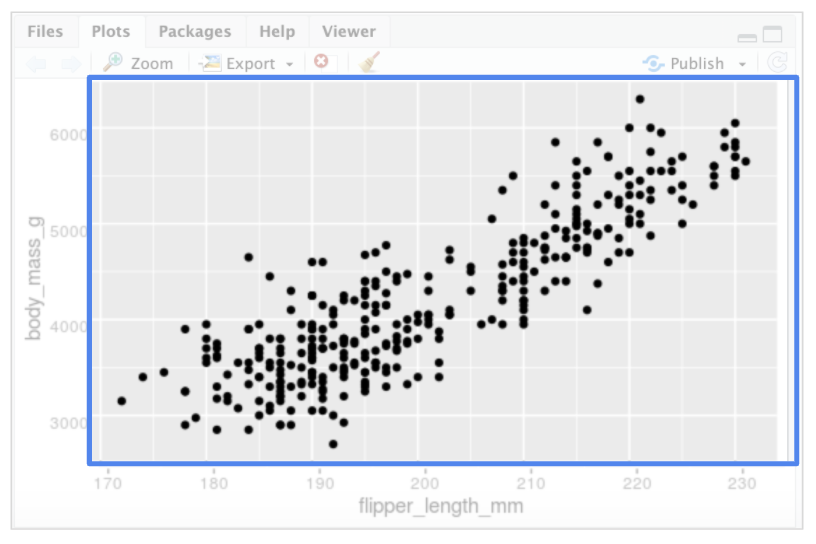
**geom\_point()**

Each geom function in ggplot2 takes a mapping argument. This defines how variables in your dataset are mapped to visual properties. The mapping argument is always paired with the aes() function. The x and y arguments of the aes() function specify which variables to map to the x-axis and the y-axis of the coordinate system. In this case, you want to map the variable “flipper\_length\_mm” to the x-axis, and the variable “body\_mass\_g” to the y-axis.

**ggplot(data = penguins) +**

**geom\_point(mapping = aes(x = flipper\_length\_mm, y = body\_mass\_g))**

Now go ahead and run the code. When you do, you get the following plot:



The plot shows a positive relationship between the two variables. In other words, the larger the penguin, the longer the flipper.

### Creating your own plot



To create your own plot using code, follow these three steps:

1. Start with the ggplot() function and choose a dataset to work with,
2. Add a geom\_ function to display your data,
3. Map the variables you want to plot in the arguments of the aes() function.

You are encouraged to try plotting with different datasets using different geoms and mapping arguments. In the upcoming videos, you’ll learn even more about the process of creating a plot. You’ll also get a chance to work with the Penguins dataset to create lots of different plots in ggplot2.

Pro-Tip: You can write the same section of code above using a different syntax with the mapping argument inside the ggplot() call:

**ggplot(data = penguins, mapping = aes(x = flipper\_length\_mm, y = body\_mass\_g)) +**

**geom\_point()**

### The ggplot2 cheat sheet



This is just the beginning of what you can do with ggplot2. If you want to find out more about ggplot2, RStudio has a useful reference guide called the “Data Visualization with ggplot2 Cheat Sheet.” You can use the Cheat Sheet as a quick reference while you work to learn about the main functions and features of ggplot2.

Click the link to check it out: [Cheat Sheet](https://ggplot2.tidyverse.org/).

## Did you complete this activity?

1 point

Yes

No