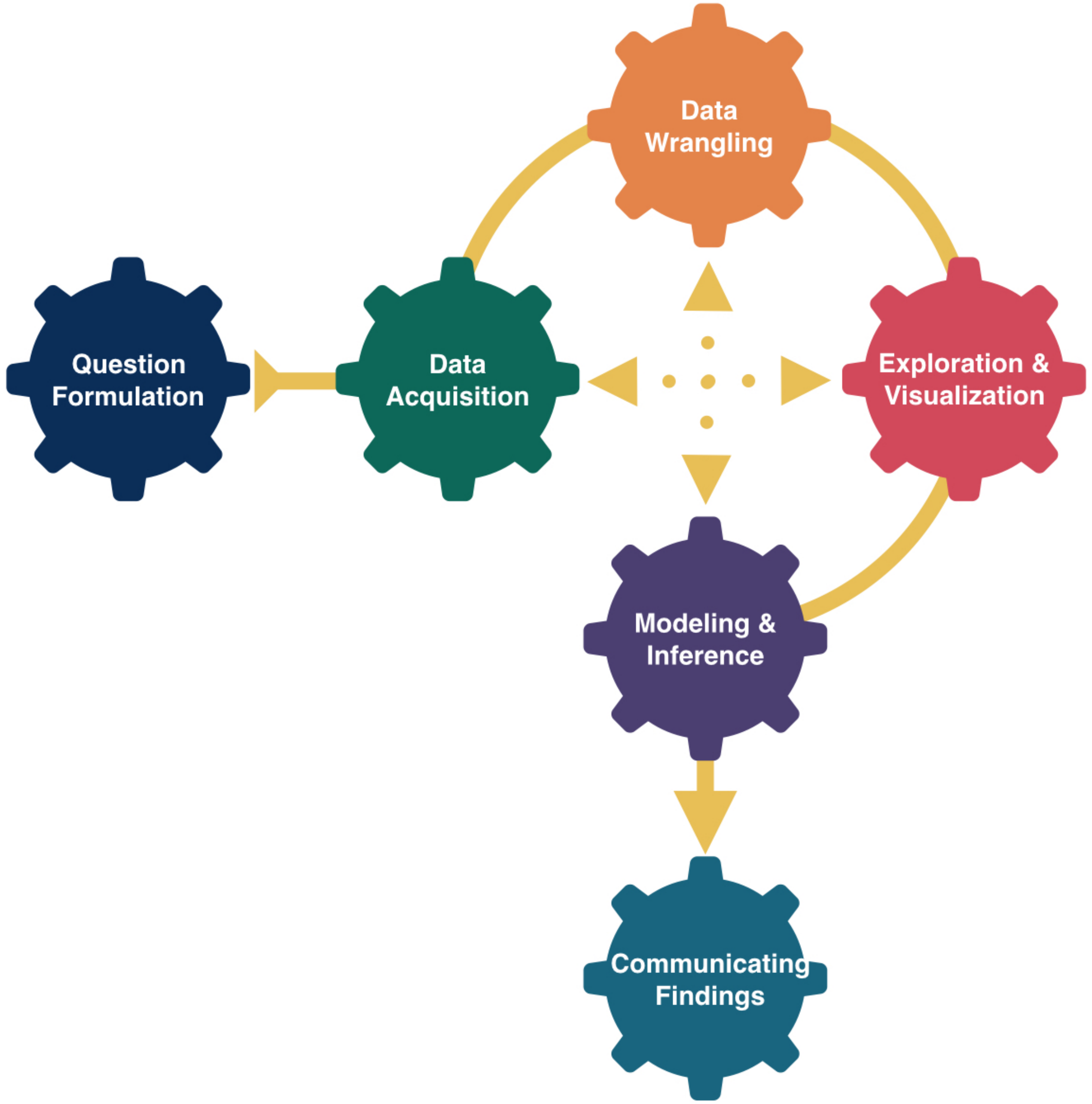


Graphing with `ggplot2`

Kelly McConville

Stat 100

Week 3 | Fall 2023

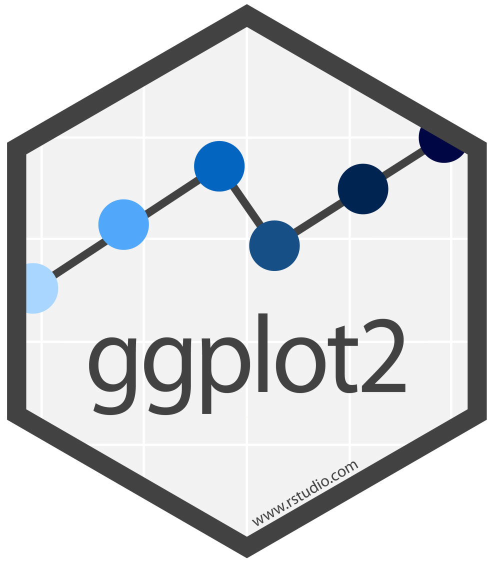
Announcements

- With COVID working its way through campus right now, make sure to check the Sections spreadsheet and the Office hours spreadsheet for updates!
- Grab a **postcard** and/or a **stamp** from SC 316 if you lost yours.
  - We also have markers, colored pencils, and crayons!
- Don’t forget that P-Set 1 due on Tuesday by 5pm in Gradescope.
- Come by office hours with any questions.

Goals for Today

- Come back to the general structure of `ggplot2`.
- Learn a few standard graphs for numerical/quantitative data:
  - **Histogram**: one numerical variable
  - **Side-by-side boxplot**: one numerical variable and one categorical variable
  - **Side-by-side violin plot**: one numerical variable and one categorical variable
  - **Scatterplot**: two numerical variables
  - **Linegraph**: two numerical variables
- And, learn the standard graphic for categorical data:
  - **Barplot**: one categorical variable
  - **Segmented barplot**: two categorical variables
- Also cover some common extensions and customizations.

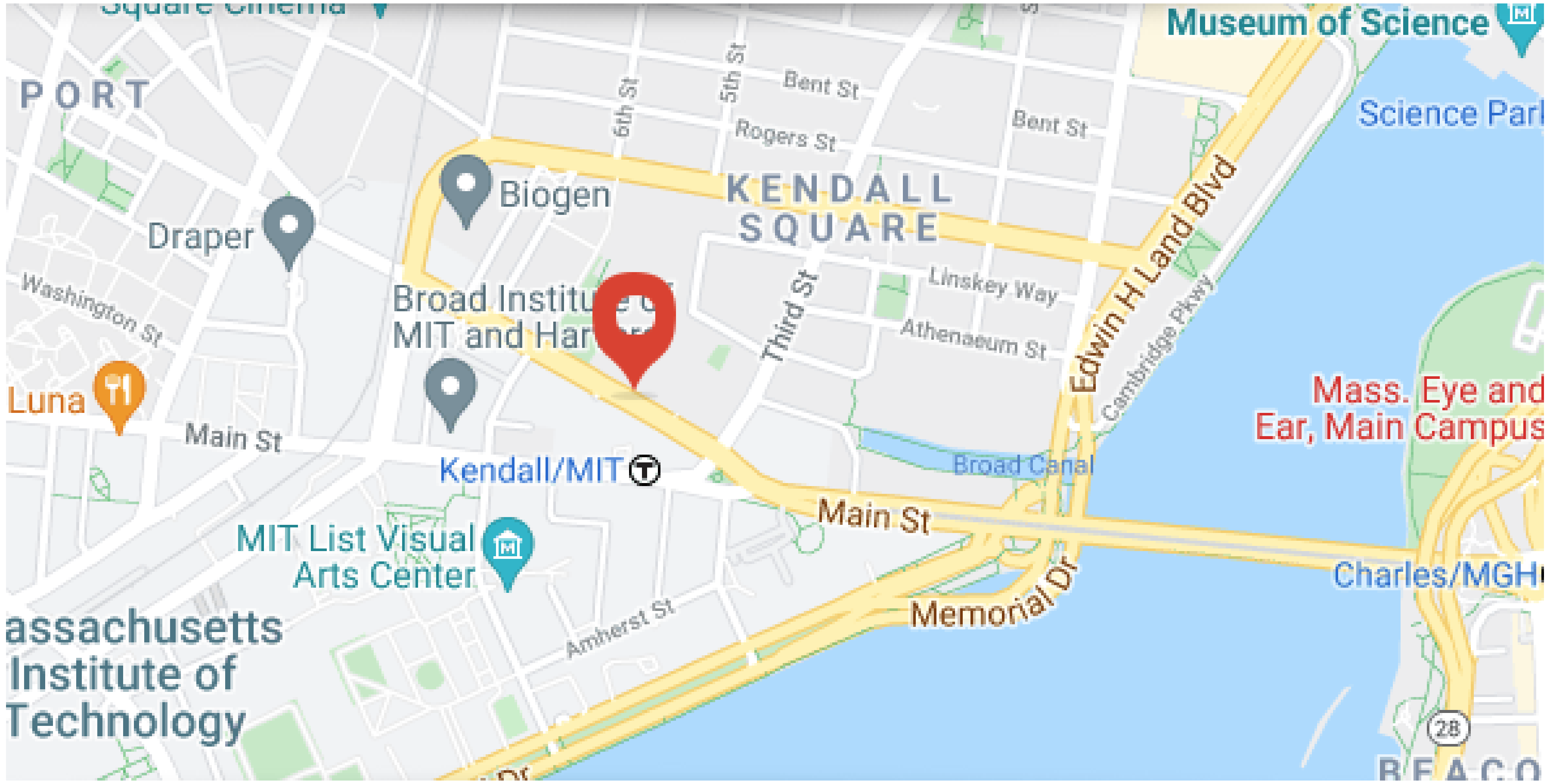
Load Necessary Packages



`ggplot2` is part of this collection of data science packages.

```r
# Load necessary packages
library(tidyverse)
```

Data Setting: Eco-Totem Broadway Bicycle Count



Import the Data

```r
july_2019 <- read_csv("data/july_2019.csv")

# Inspect the data
glimpse(july_2019)
```

```
Rows: 192
Columns: 8
$ DateTime  <chr> "07/04/2019 12:00:00 AM", "07/04/2019 12:15:00 AM", "07/04/2…
$ Day       <chr> "Thursday", "Thursday", "Thursday", "Thursday", "Thursday", …
$ Date      <date> 2019-07-04, 2019-07-04, 2019-07-04, 2019-07-04, 2019-07-04,…
$ Time      <time> 00:00:00, 00:15:00, 00:30:00, 00:45:00, 01:00:00, 01:15:00,…
$ Total     <dbl> 2, 3, 2, 0, 3, 2, 1, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 1, 1, …
$ Westbound <dbl> 2, 3, 1, 0, 2, 2, 1, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0, 1, …
$ Eastbound <dbl> 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, …
$ Occasion  <chr> "Fourth of July", "Fourth of July", "Fourth of July", "Fourt…
```

ggplot2 example code

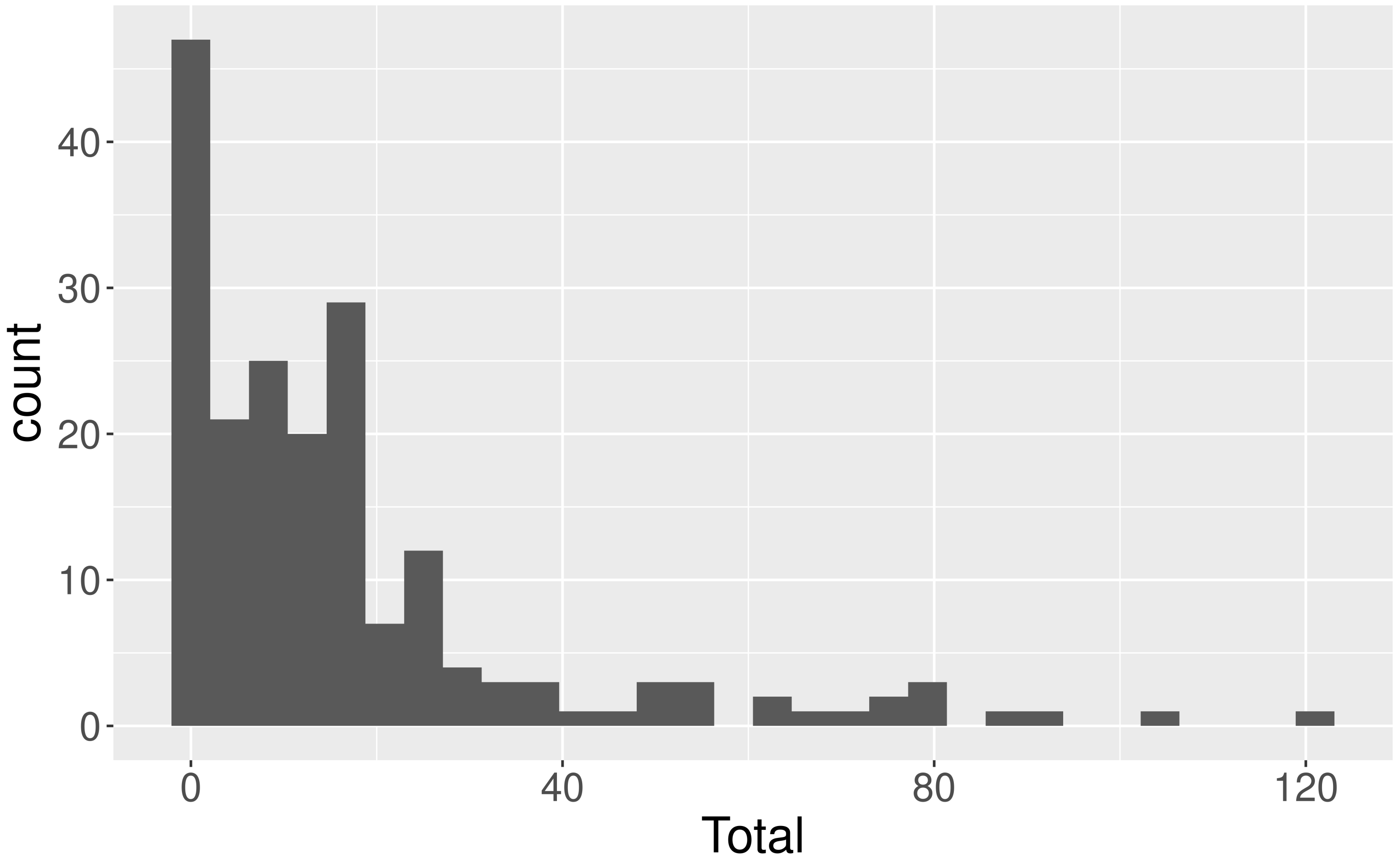
Guiding Principle: We will map variables from the data to the aesthetic attributes (e.g. location, size, shape, color) of geometric objects (e.g. points, lines, bars).

```r
ggplot(data = ---, mapping = aes(---)) +
  geom_---(---)
```

There are other layers, such as `scales_---_---()` and `labs()`, but we will wait on those.

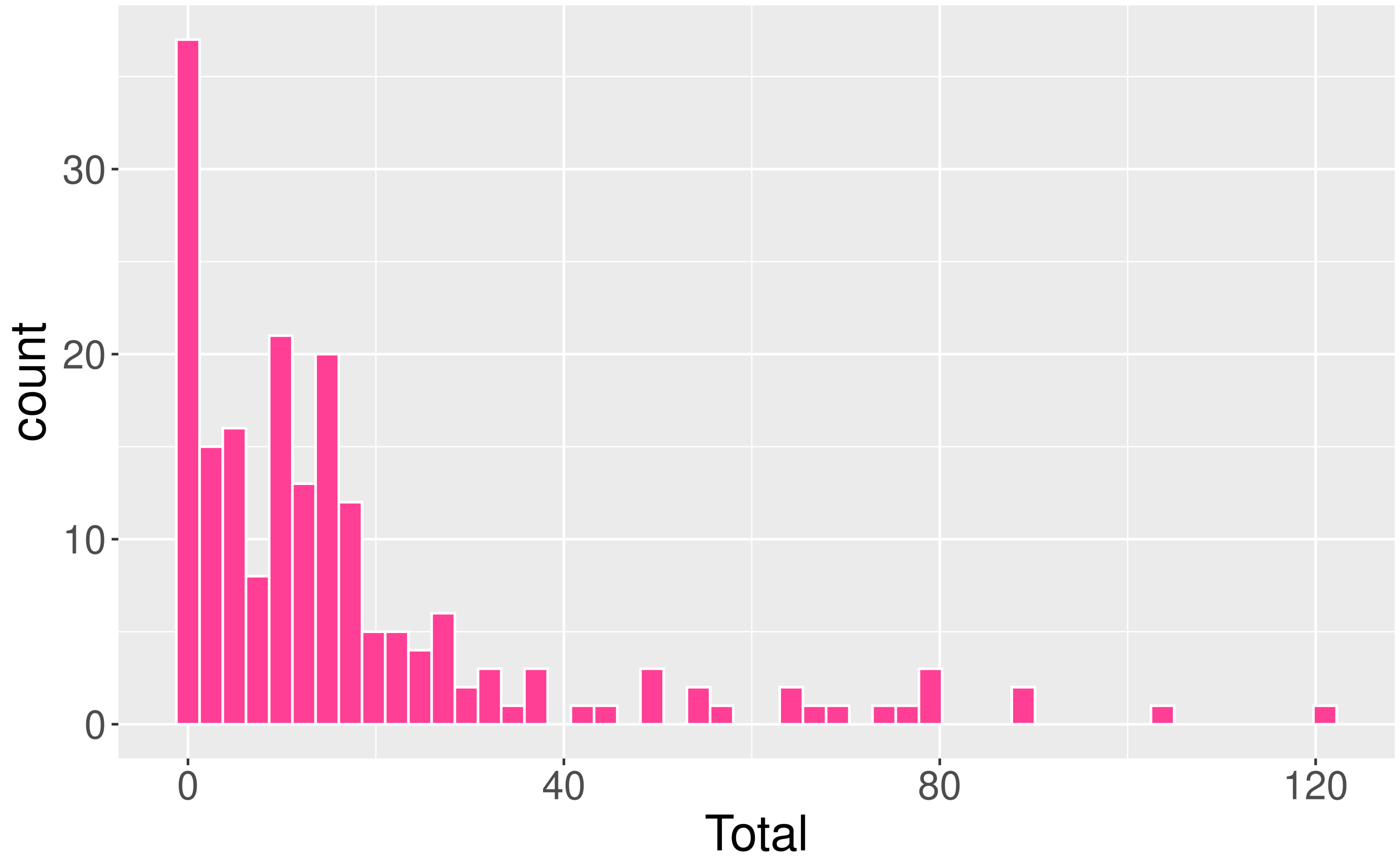
Histograms

```r
# Create histogram
ggplot(data = july_2019,
       mapping = aes(x = Total)) +
  geom_histogram()
```



Histograms

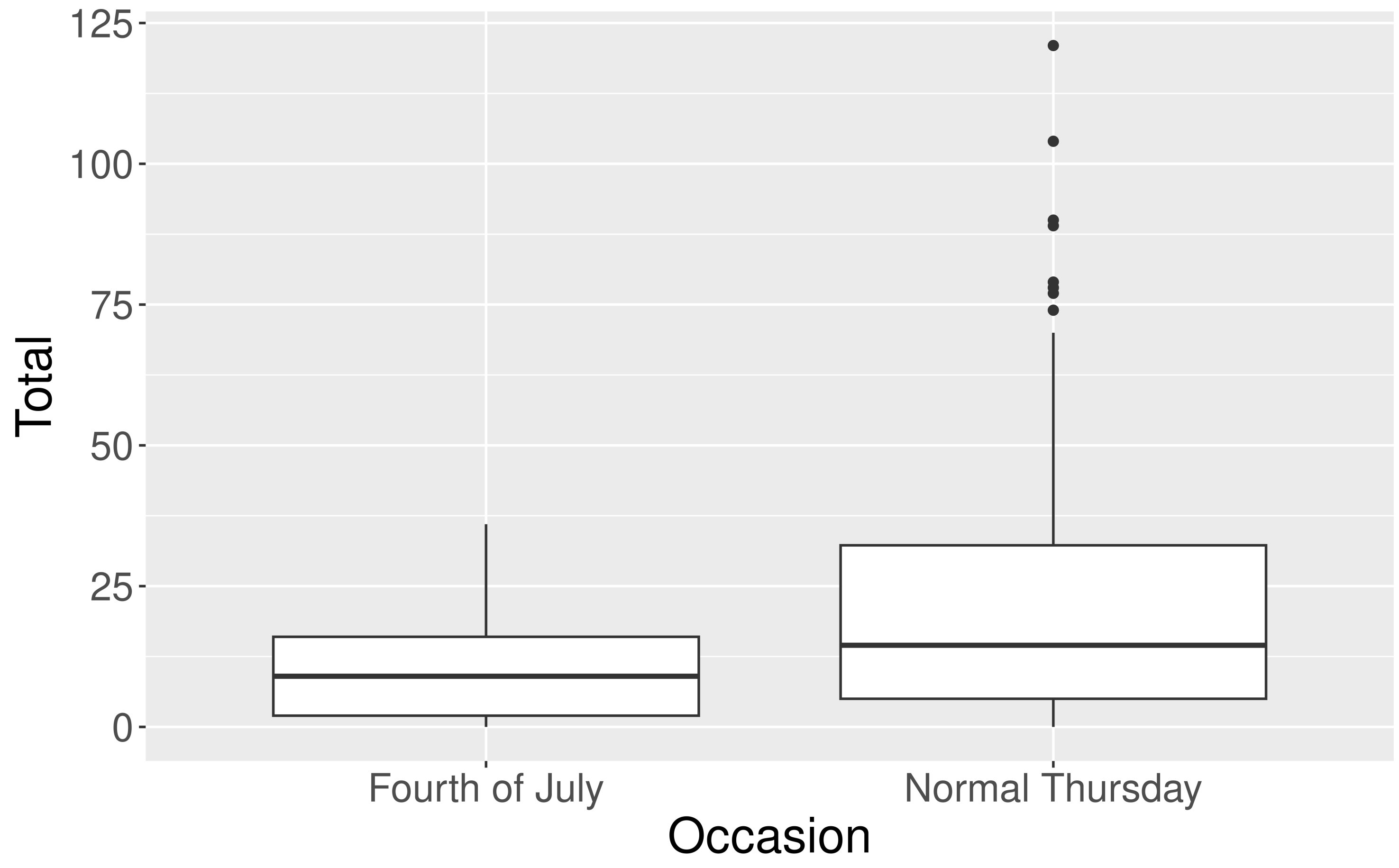
```r
# Create histogram
ggplot(data = july_2019,
       mapping = aes(x = Total)) +
  geom_histogram(color = "white",
                 fill = "violetred1",
                 bins = 50)
```



- mapping to a variable goes in `aes()`
- setting to a specific value goes in the `geom_---()`

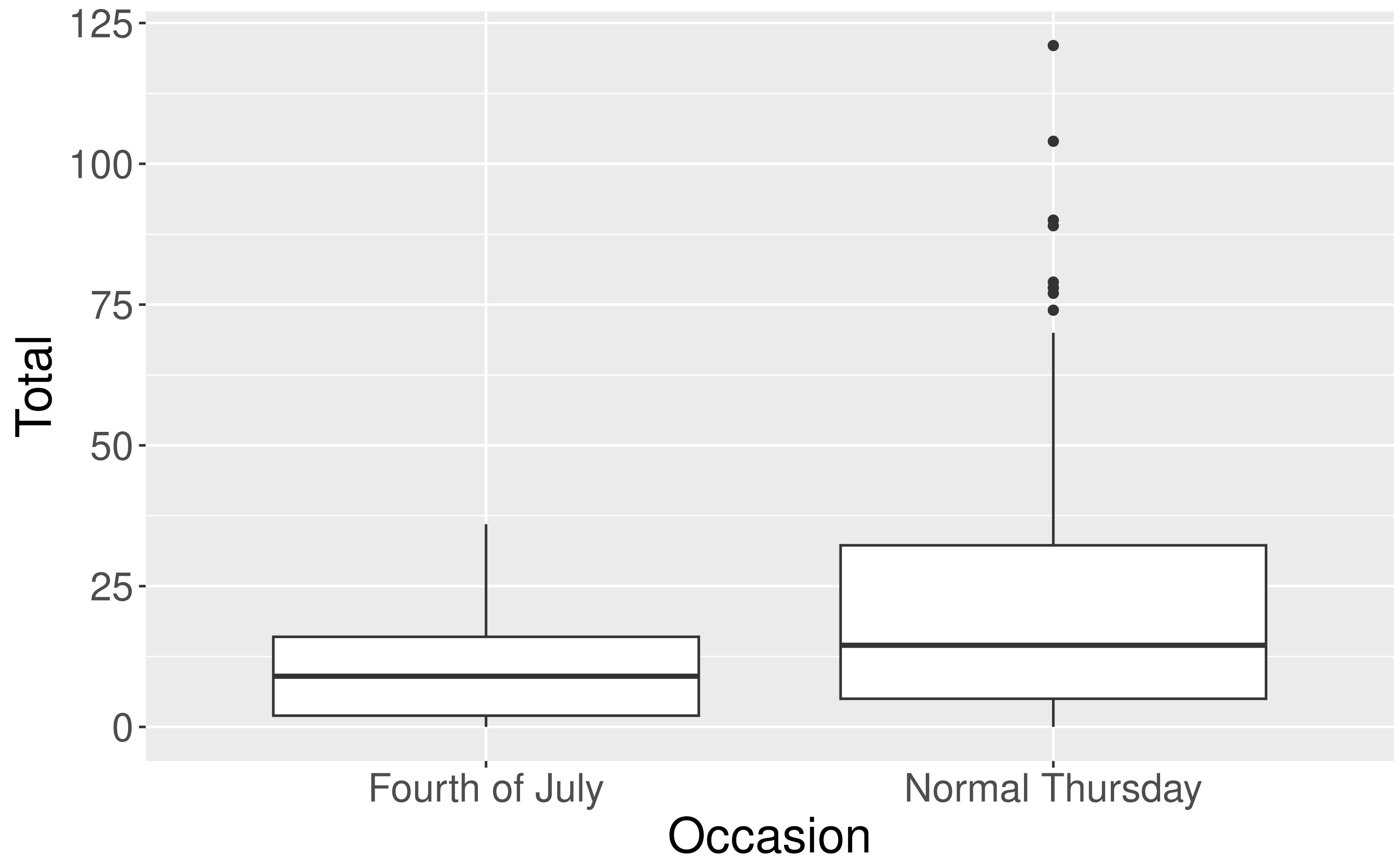
Boxplots

- Five number summary:
  - Minimum
  - First quartile (Q1)
  - Median
  - Third quartile (Q3)
  - Maximum
- Interquartile range (IQR) = Q3 – Q1
- Outliers: unusual points
  - Boxplot defines unusual as being beyond $1.5 * IQR$ from $Q1$ or $Q3$.
- Whiskers: reach out to the furthest point that is NOT an outlier



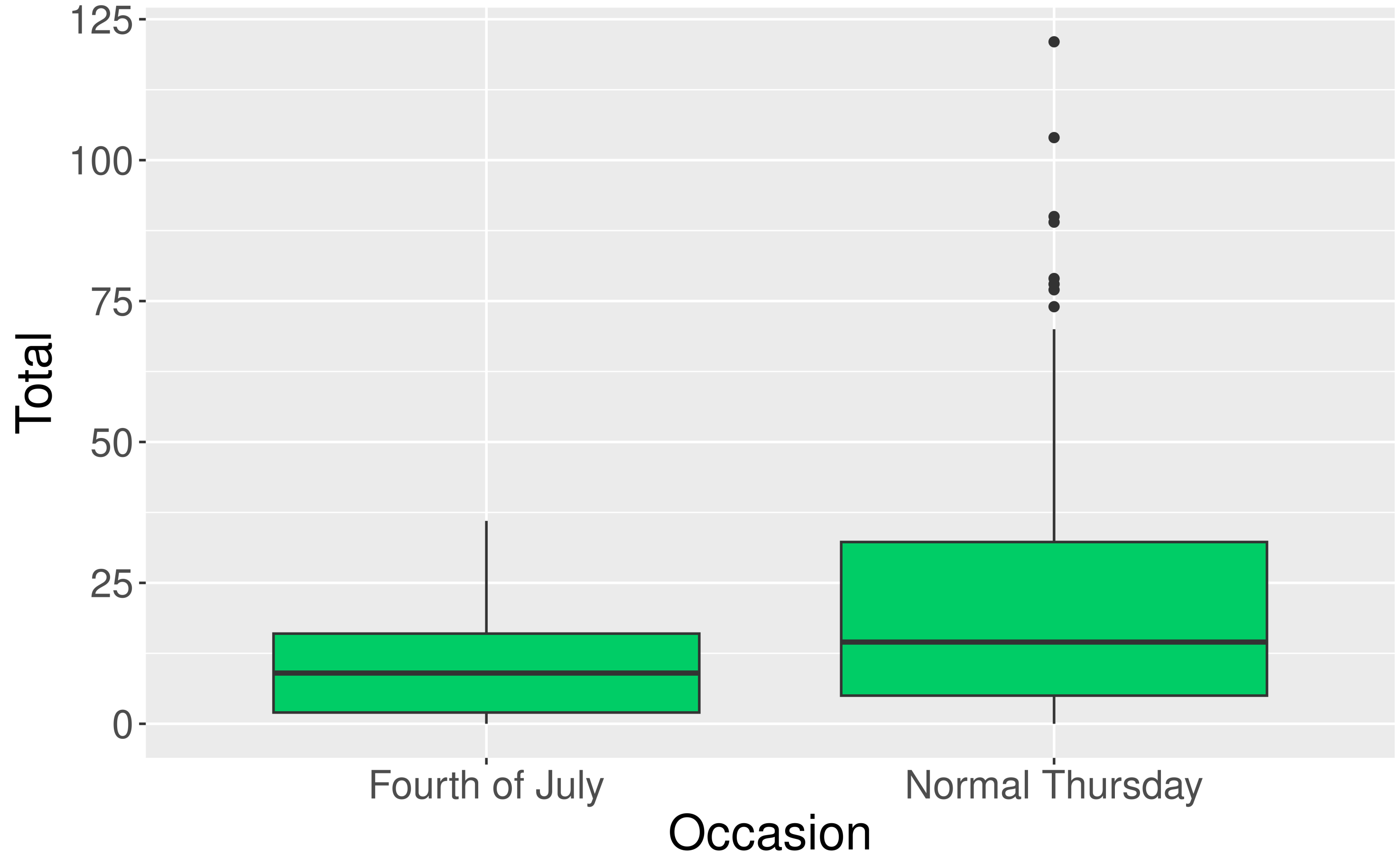
Boxplots

```
# Create boxplot
ggplot(data = july_2019,
       mapping = aes(x = Occasion,
                     y = Total)) +
  geom_boxplot()
```



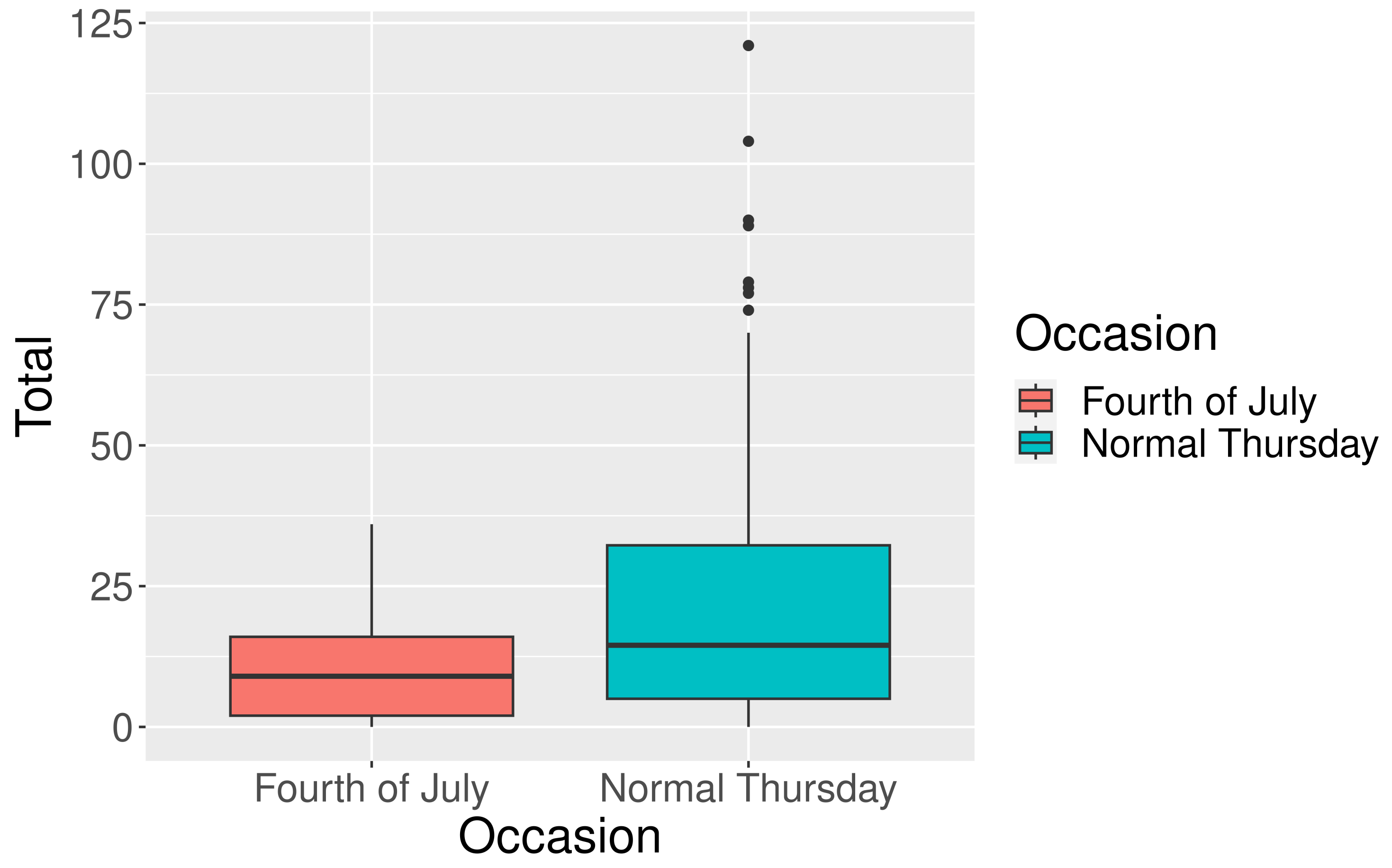
Boxplots

```
ggplot(data = july_2019,
       mapping = aes(x = Occasion,
                     y = Total)) +
  geom_boxplot(fill = "springgreen3")
```



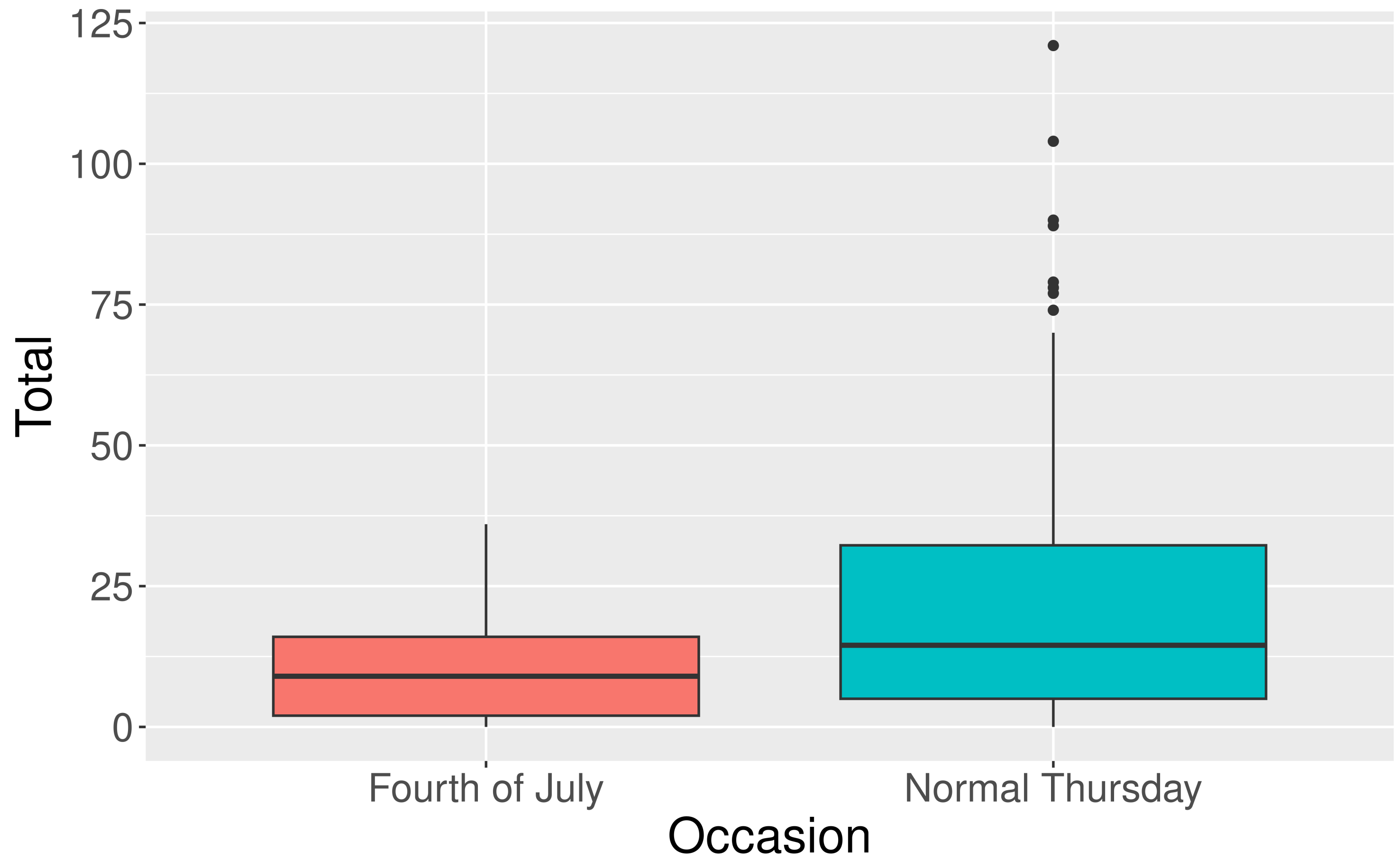
Boxplots

```r
ggplot(data = july_2019,
       mapping = aes(x = Occasion,
                     y = Total,
                     fill = Occasion)) +
  geom_boxplot()
```



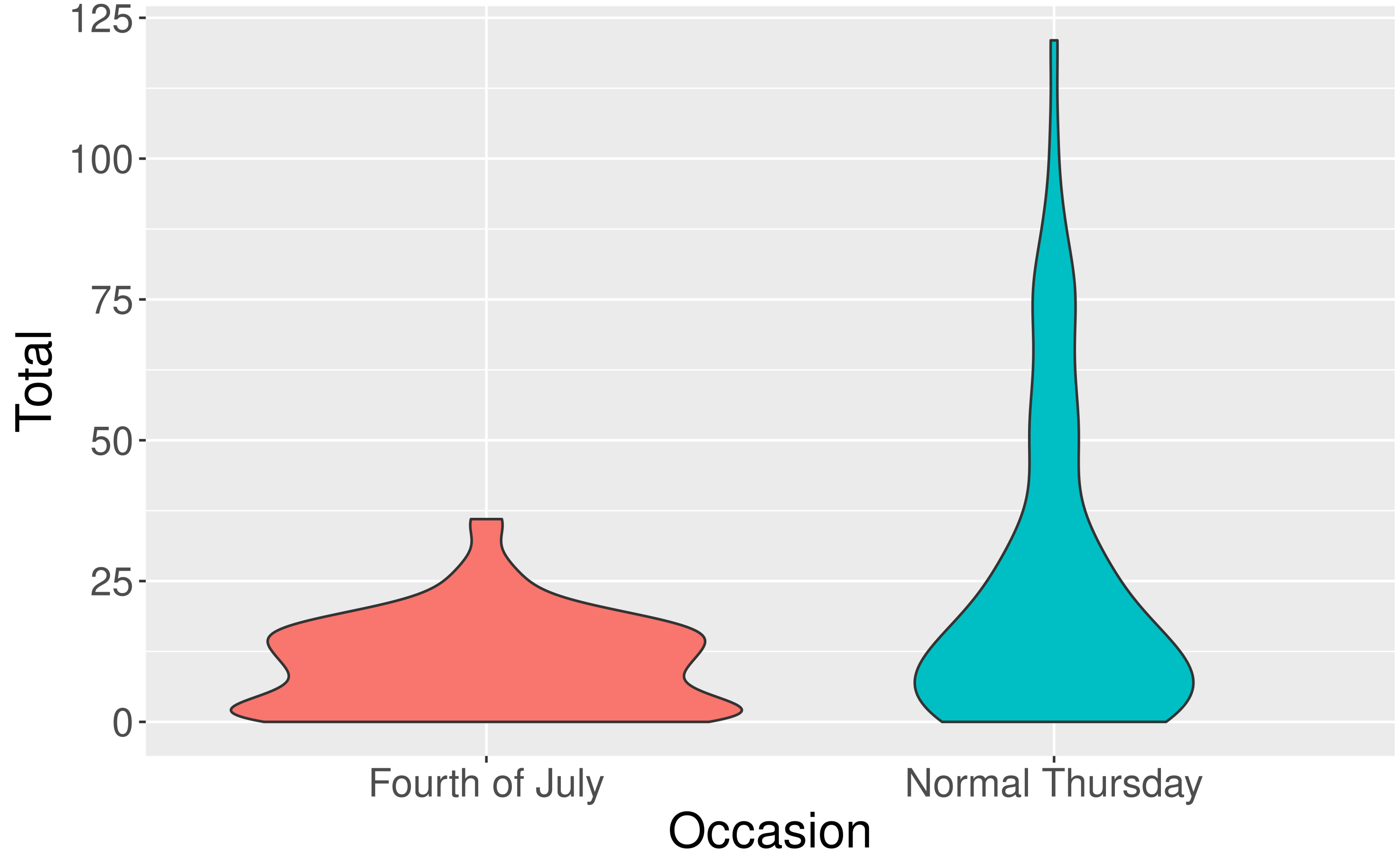
Boxplots

```
ggplot(data = july_2019,
       mapping = aes(x = Occasion,
                     y = Total,
                     fill = Occasion)) +
  geom_boxplot() +
  guides(fill = "none")
```

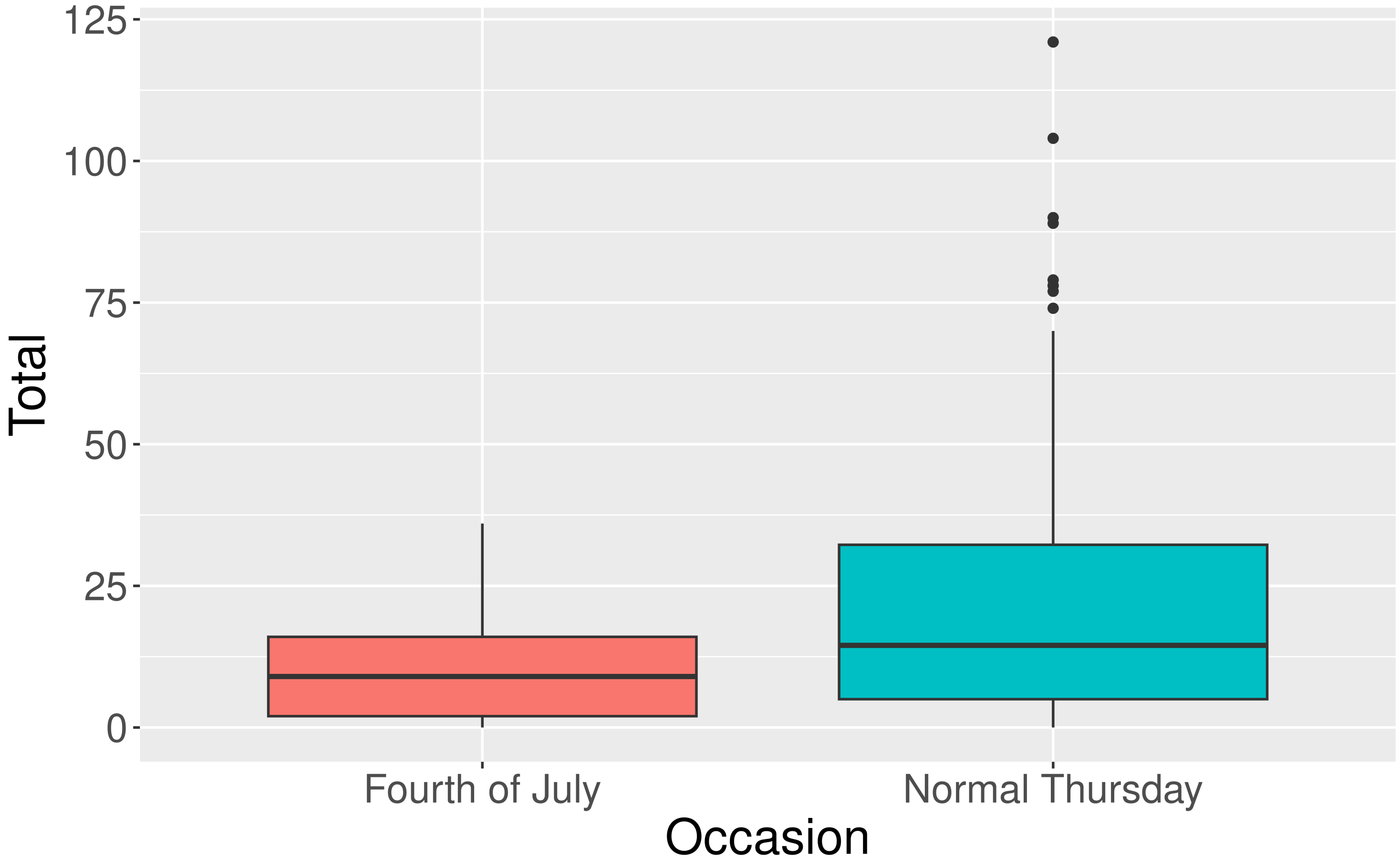


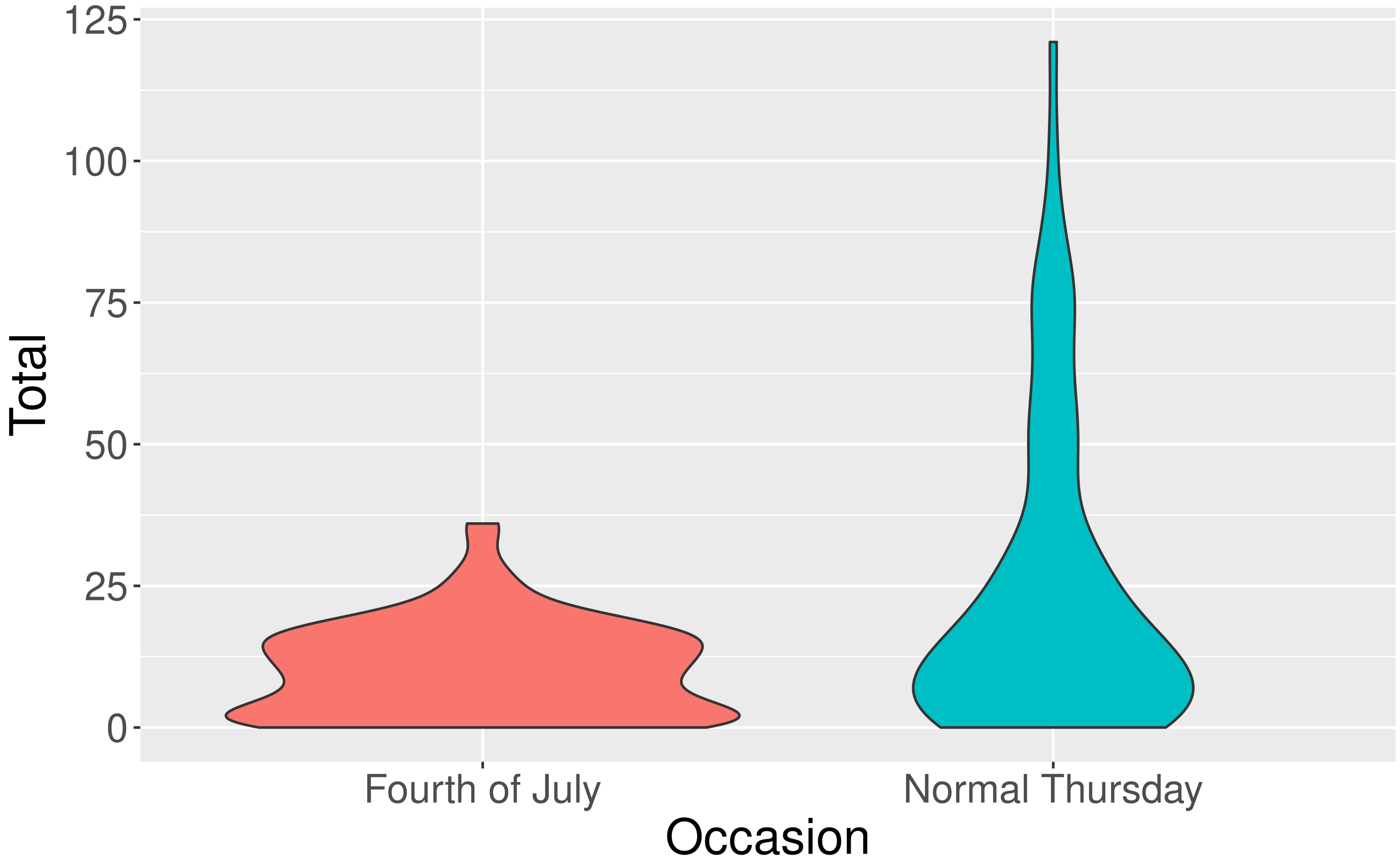
Violin Plots

```r
ggplot(data = july_2019,
       mapping = aes(x = Occasion,
                     y = Total,
                     fill = Occasion)) +
  geom_violin() +
  guides(fill = "none")
```



Boxplot Versus Violin Plots

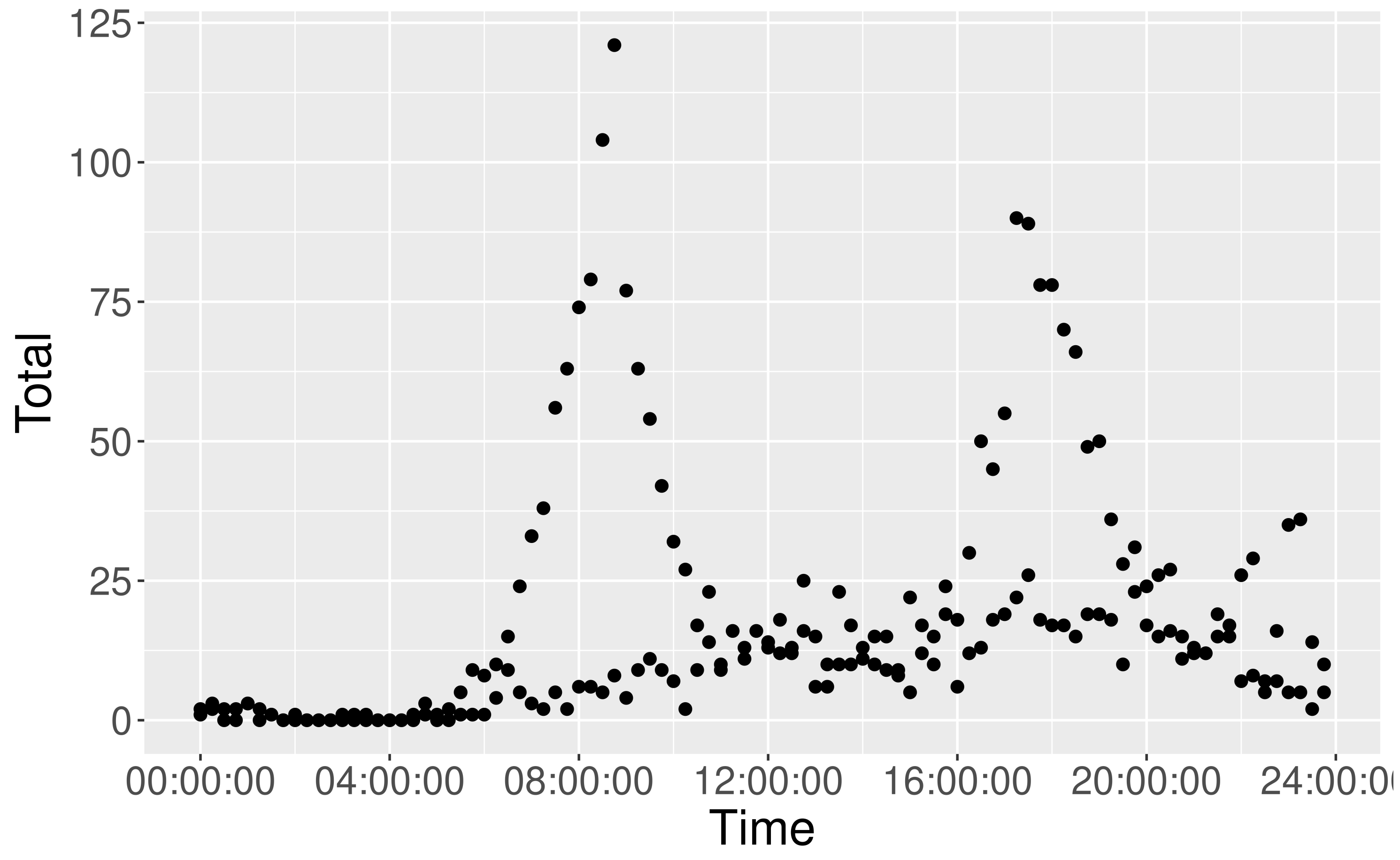




Scatterplots

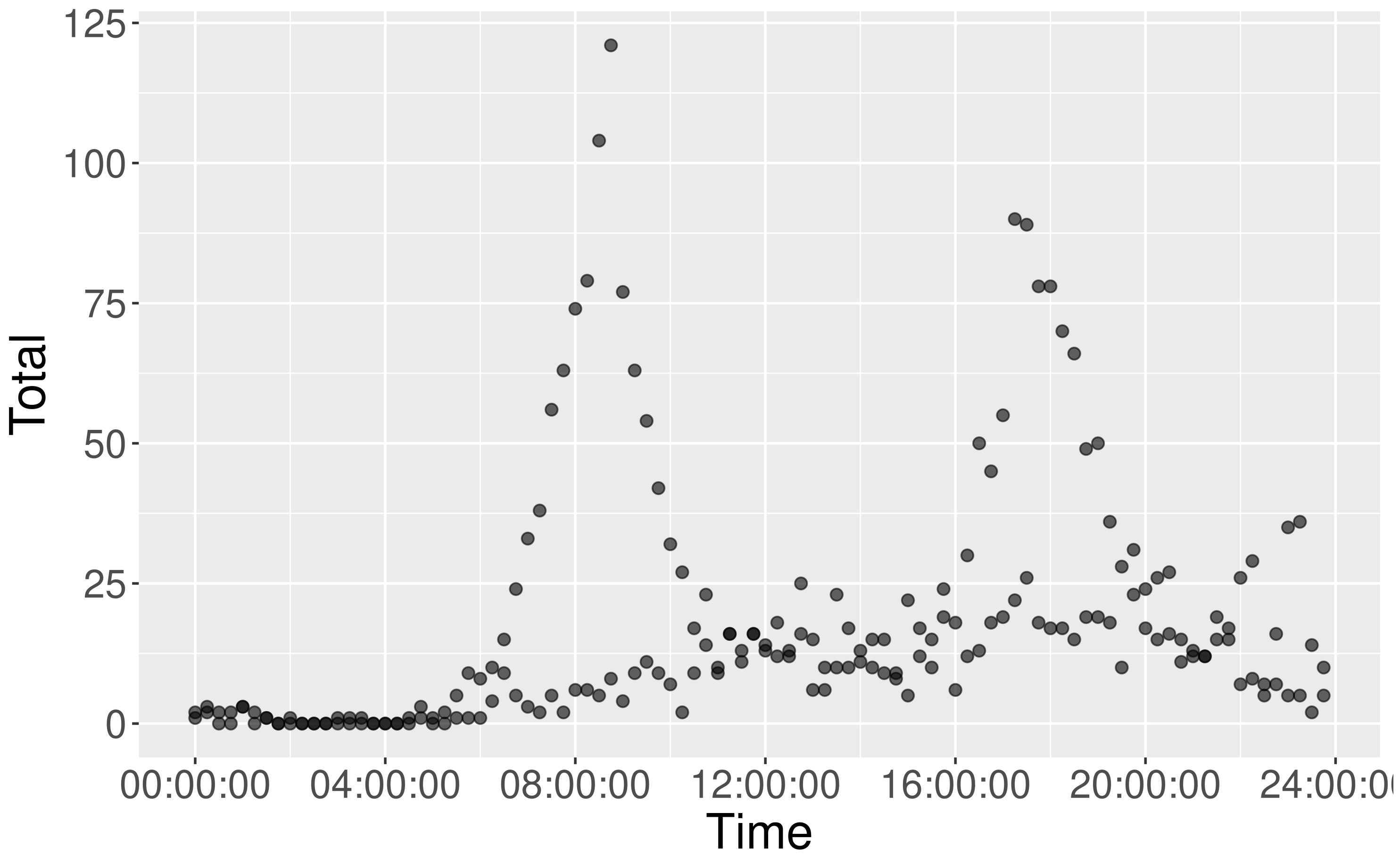
- Explore relationships between numerical variables.
  - We will be especially interested in linear relationships.

```r
ggplot(data = july_2019,
       mapping = aes(x = Time,
                     y = Total)) +
  geom_point(size = 2)
```



Scatterplots

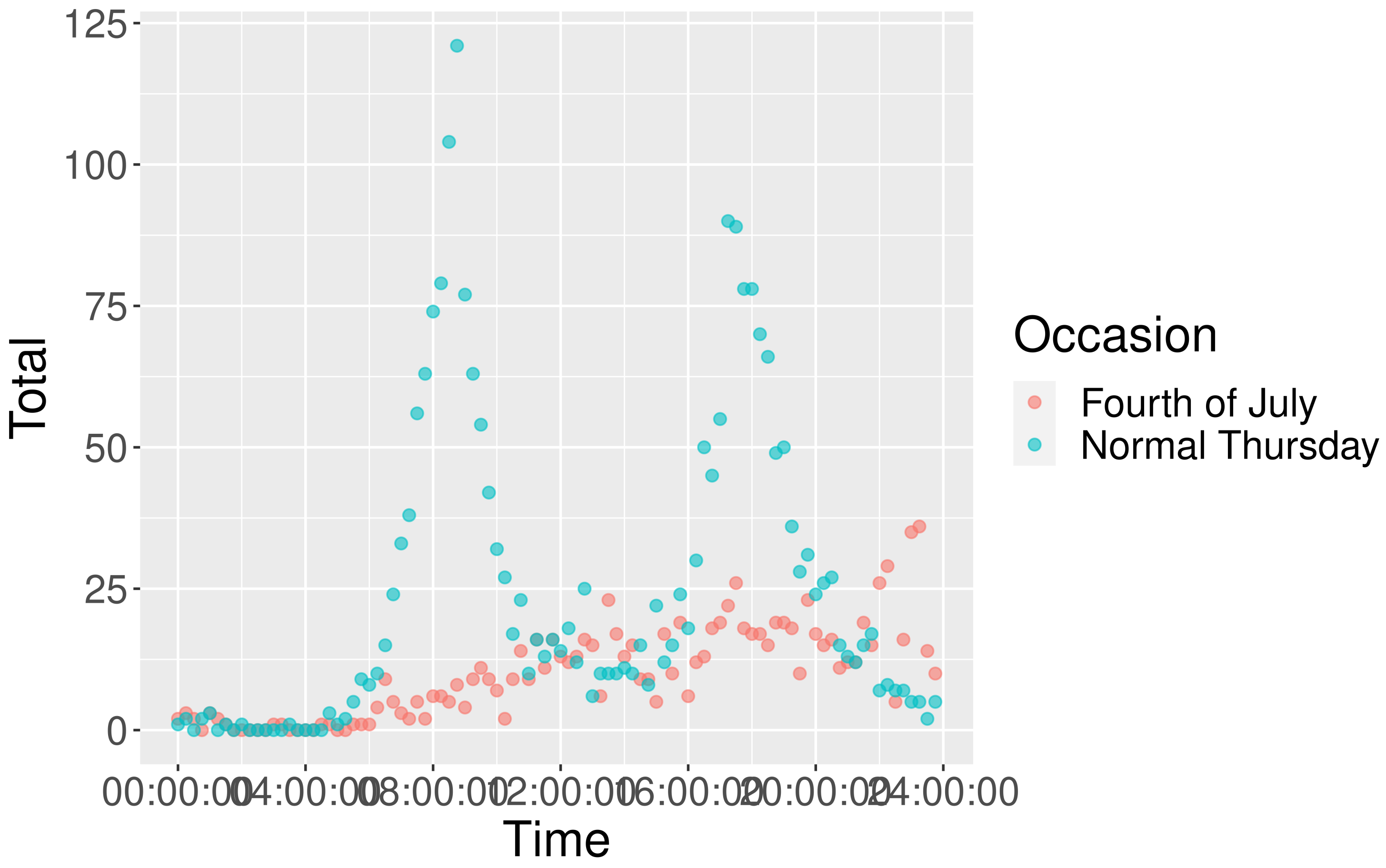
```r
ggplot(data = july_2019,
       mapping = aes(x = Time,
                     y = Total)) +
  geom_point(size = 2, alpha = 0.6)
```



- Fix over-plotting
- Why the weird pattern??

Scatterplots

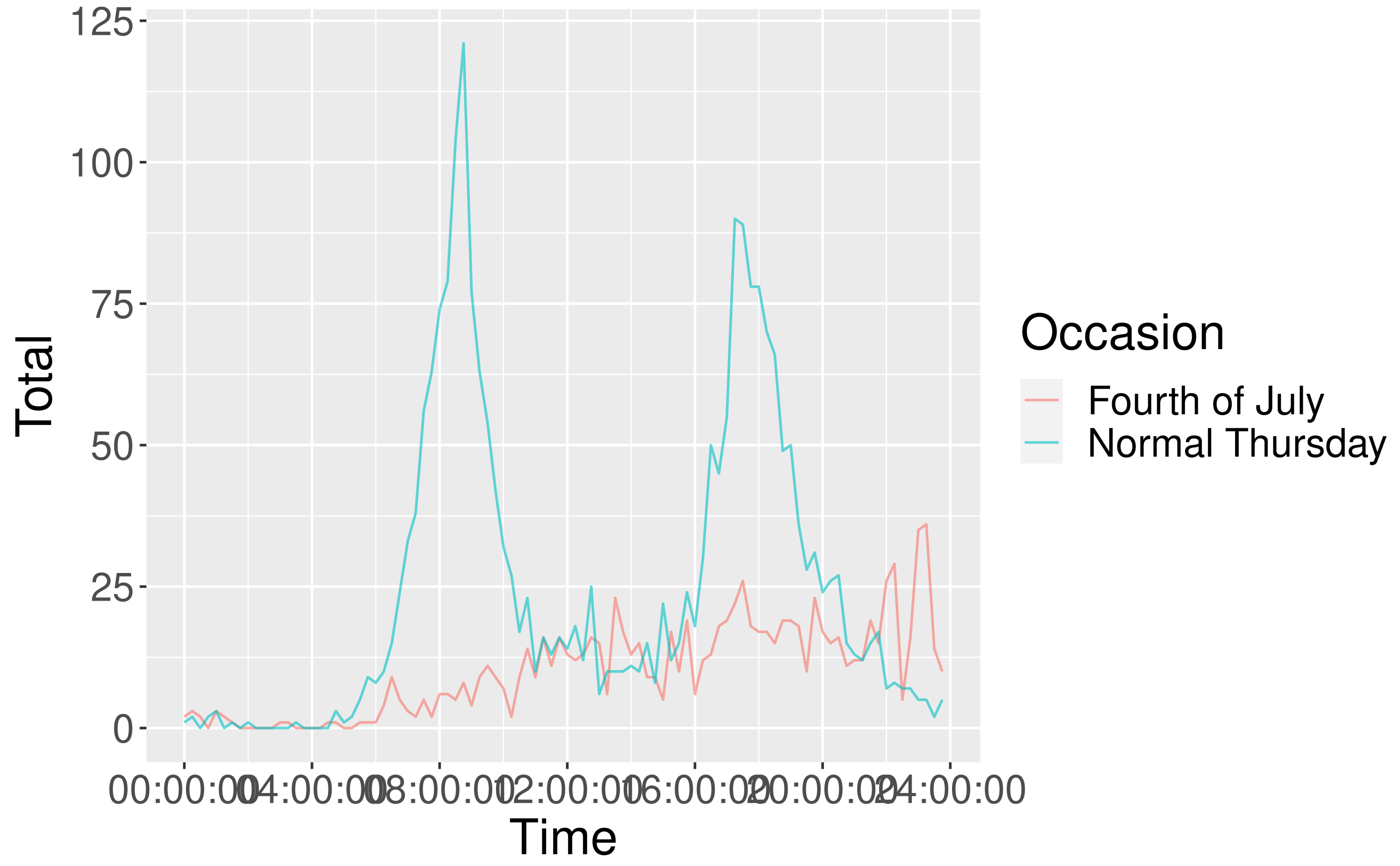
```r
ggplot(data = july_2019,
       mapping = aes(x = Time,
                     y = Total,
                     color = Occasion)) +
  geom_point(size = 2, alpha = 0.6)
```



Linegraphs

Also called time series plot when time is represented on the x axis.

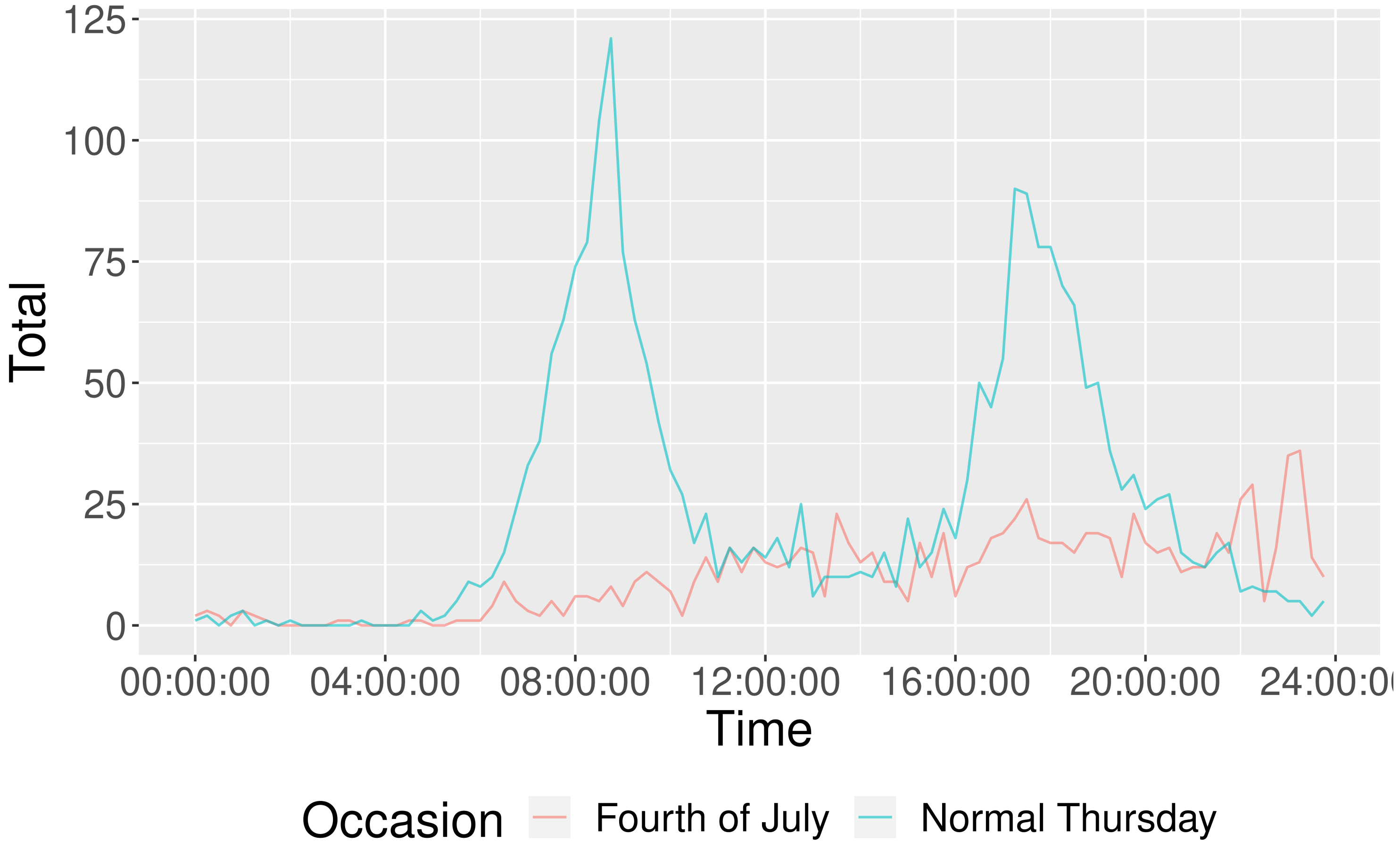
```r
ggplot(data = july_2019,
       mapping = aes(x = Time,
                     y = Total,
                     color = Occasion)) +
  geom_line(alpha = 0.6)
```



Linegraphs

Also called **time series plot** when time is represented on the x axis.

```r
ggplot(data = july_2019,
       mapping = aes(x = Time,
                     y = Total,
                     color = Occasion)) +
  geom_line(alpha = 0.6) +
  theme(legend.pos = "bottom")
```



New Data Setting: Dog Names in Cambridge, MA

Based on dog license data collected by Cambridge's Animal Commission

```r
# Import and inspect data
dogs <- read_csv("https://data.cambridgema.gov/api/views/sckh-3xyx/rows.csv")
glimpse(dogs)
```

```
Rows: 3,942
Columns: 6
$ Dog_Name          <chr> "Butch", "Baxter", "Bodhi", "Ocean", "Coco", "Brio", …
$ Dog_Breed         <chr> "Mixed Breed", "Mixed Breed", "Golden Retriever", "Pu…
$ Location_masked   <chr> "POINT (-71.1328 42.3989)", "POINT (-71.1186 42.3814)…
$ Latitude_masked   <dbl> 42.3989, 42.3814, 42.3998, 42.3726, 42.3610, 42.3892,…
$ Longitude_masked <dbl> -71.1328, -71.1186, -71.1308, -71.1087, -71.1022, -71…
$ Neighborhood      <chr> "North Cambridge", "Neighborhood Nine", "North Cambri…
```

Data Wrangling

We haven’t learned this topic yet.

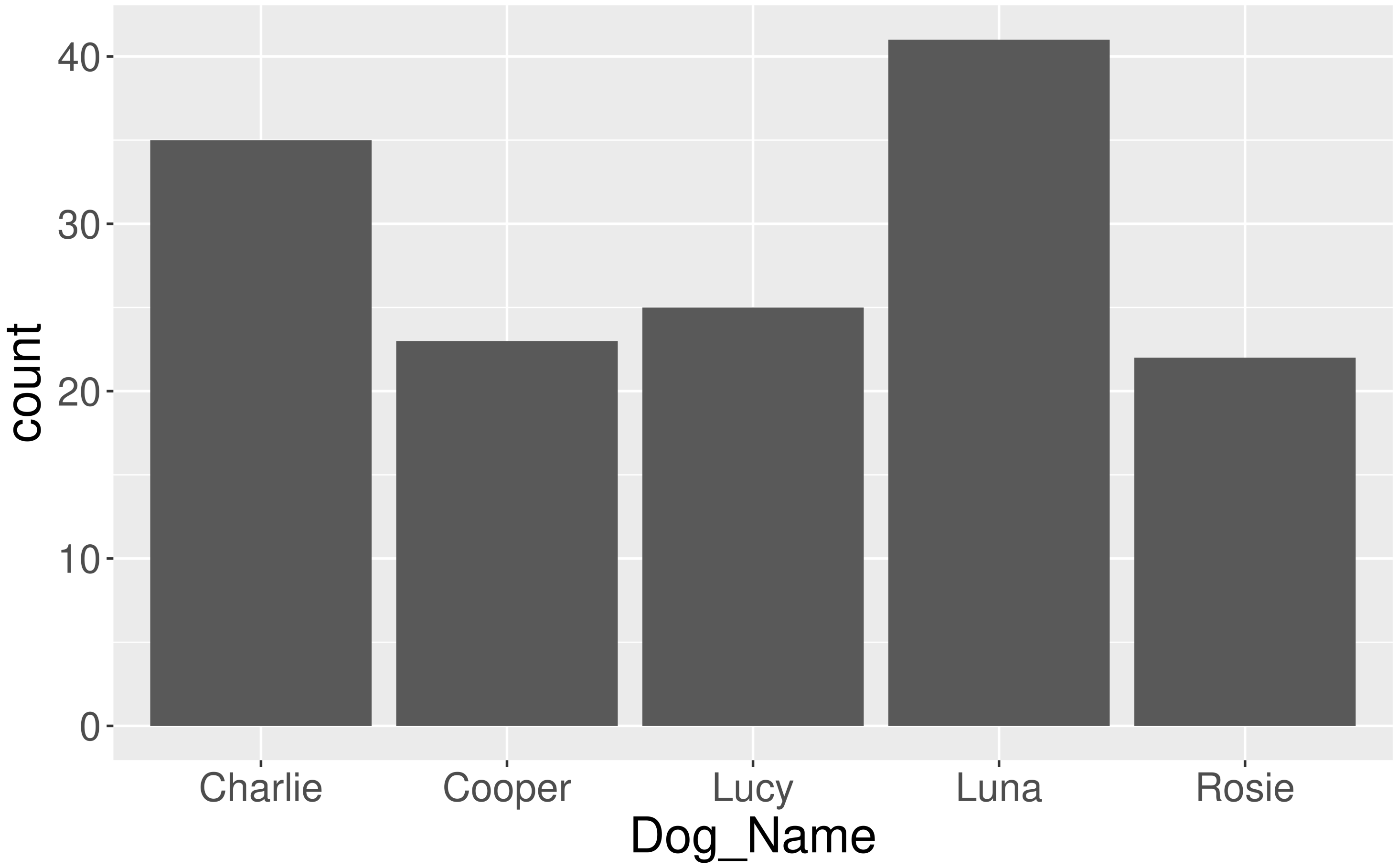
I only included this code for completeness/transparency.

```r
# Create a column for Breed
dogs <- mutate(dogs, Breed = if_else(
                       Dog_Breed == "Mixed Breed",
                       "Mixed", "Single"))


# Find the 5 top most common names
top5names <- count(dogs, Dog_Name) %>%
  slice_max(n = 5, order_by = n) %>%
  select(Dog_Name) %>%
  pull()

# Filter dataset to only the 5 top most common names
dogs_top5 <- filter(dogs,
                     Dog_Name %in% top5names)
```

Before we graph the data, do we have any guesses on popular dog names?

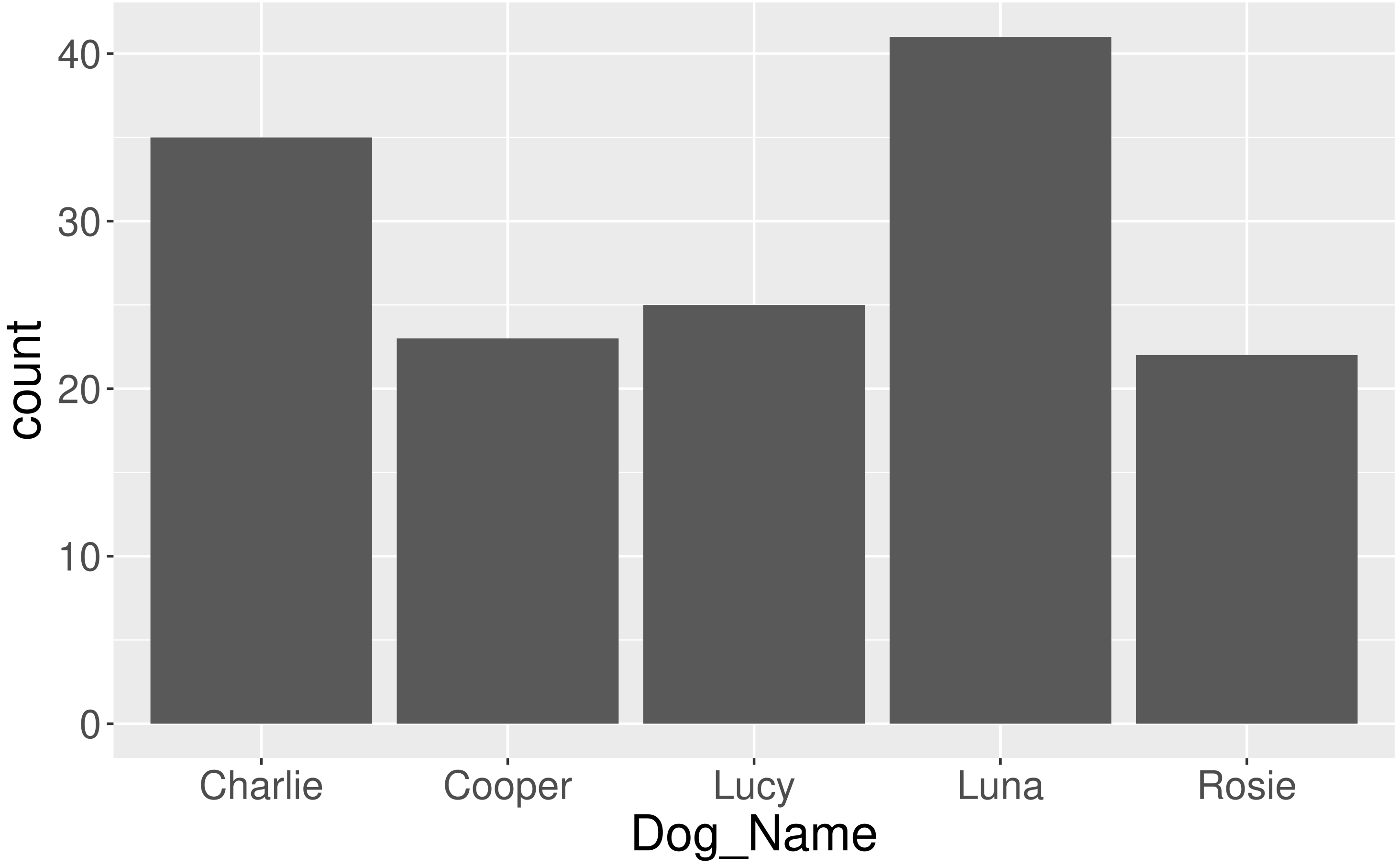
Barplots

Displays the frequency for each category.



Barplots

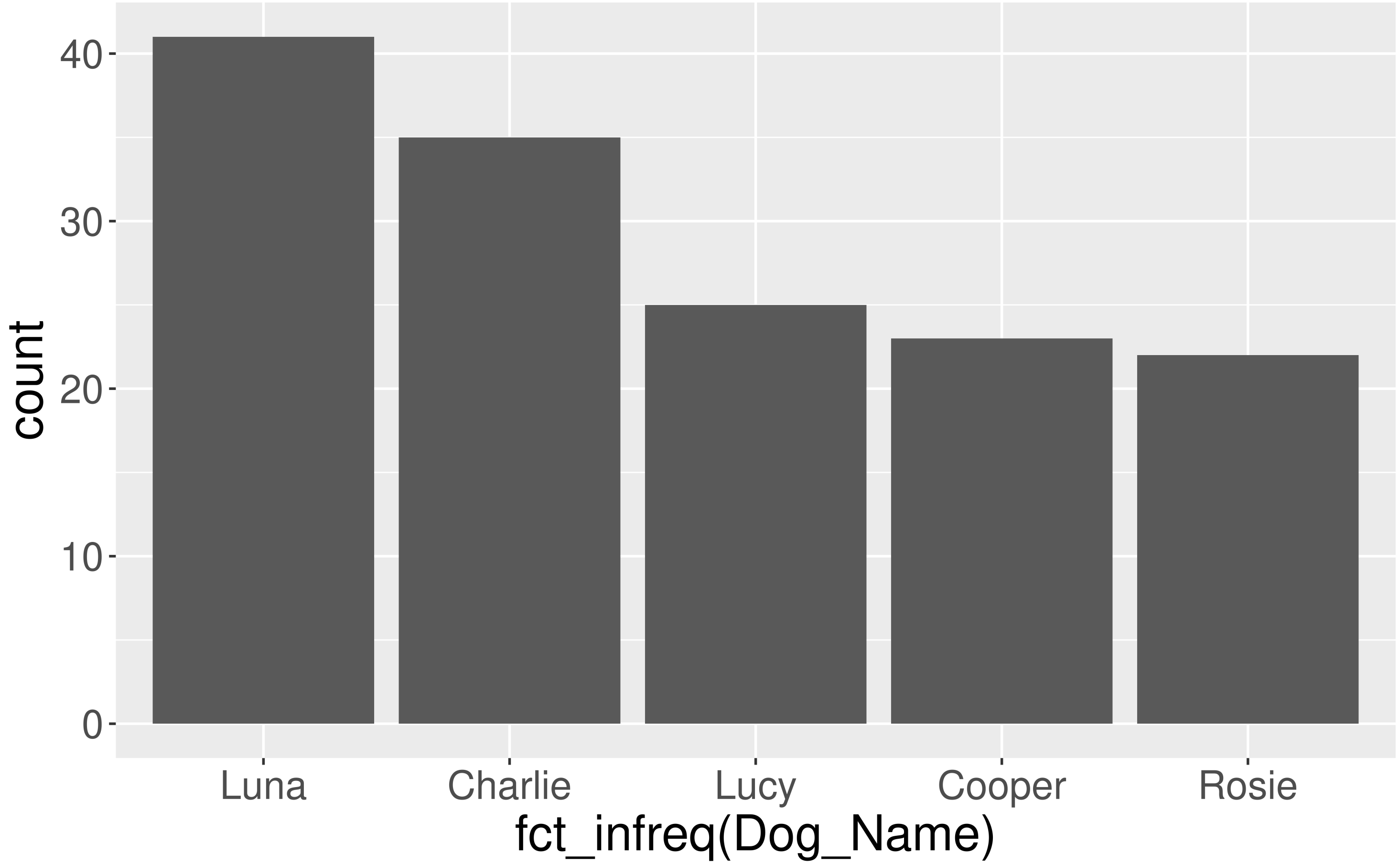
```r
# Create barplot
ggplot(data = dogs_top5,
     mapping = aes(x = Dog_Name)) +
  geom_bar()
```



How could we make this graph better?

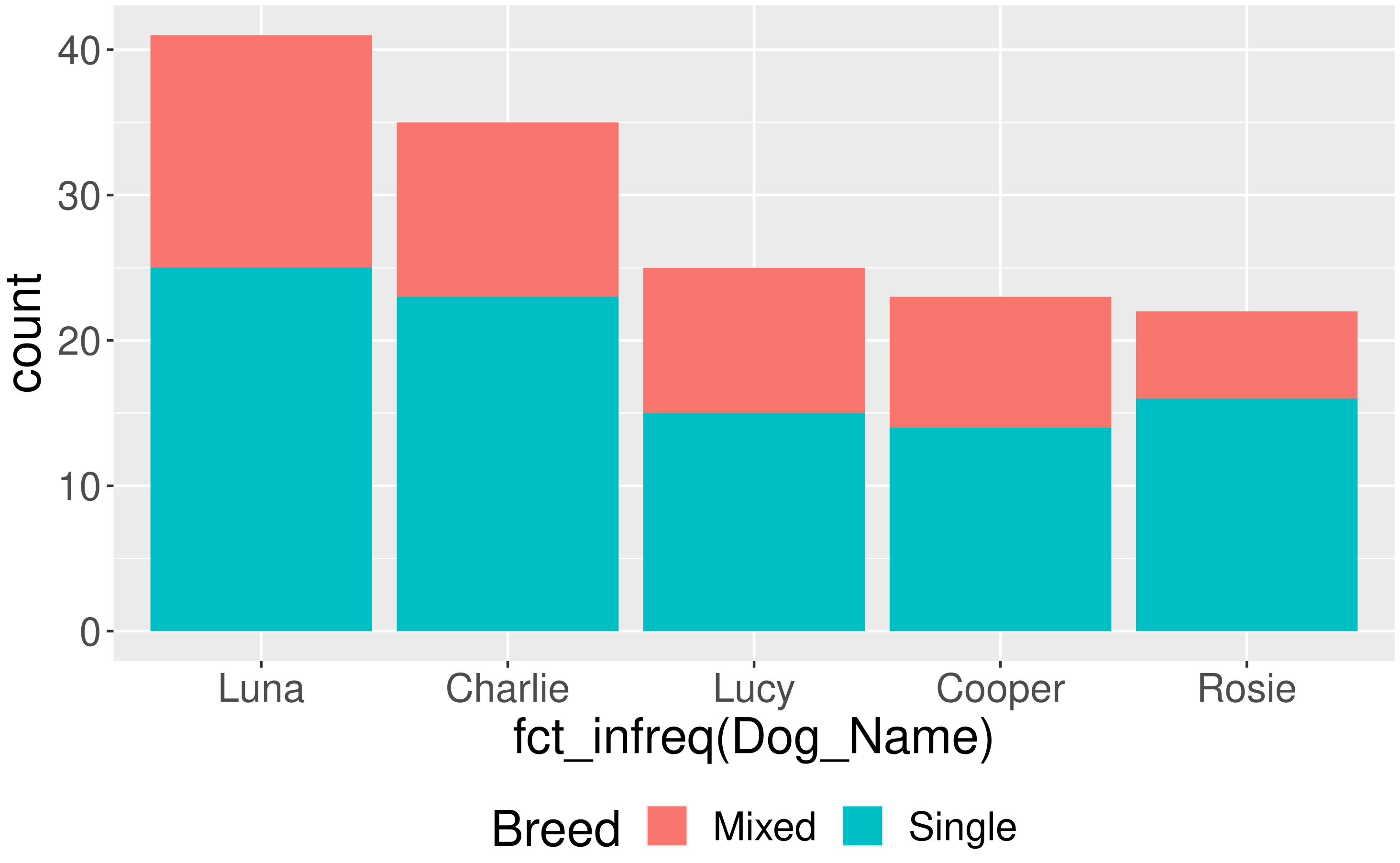
Barplots

```r
# Create barplot
ggplot(data = dogs_top5,
  mapping = aes(x = fct_infreq(Dog_Name))) +
  geom_bar()
```



Segmented Barplots

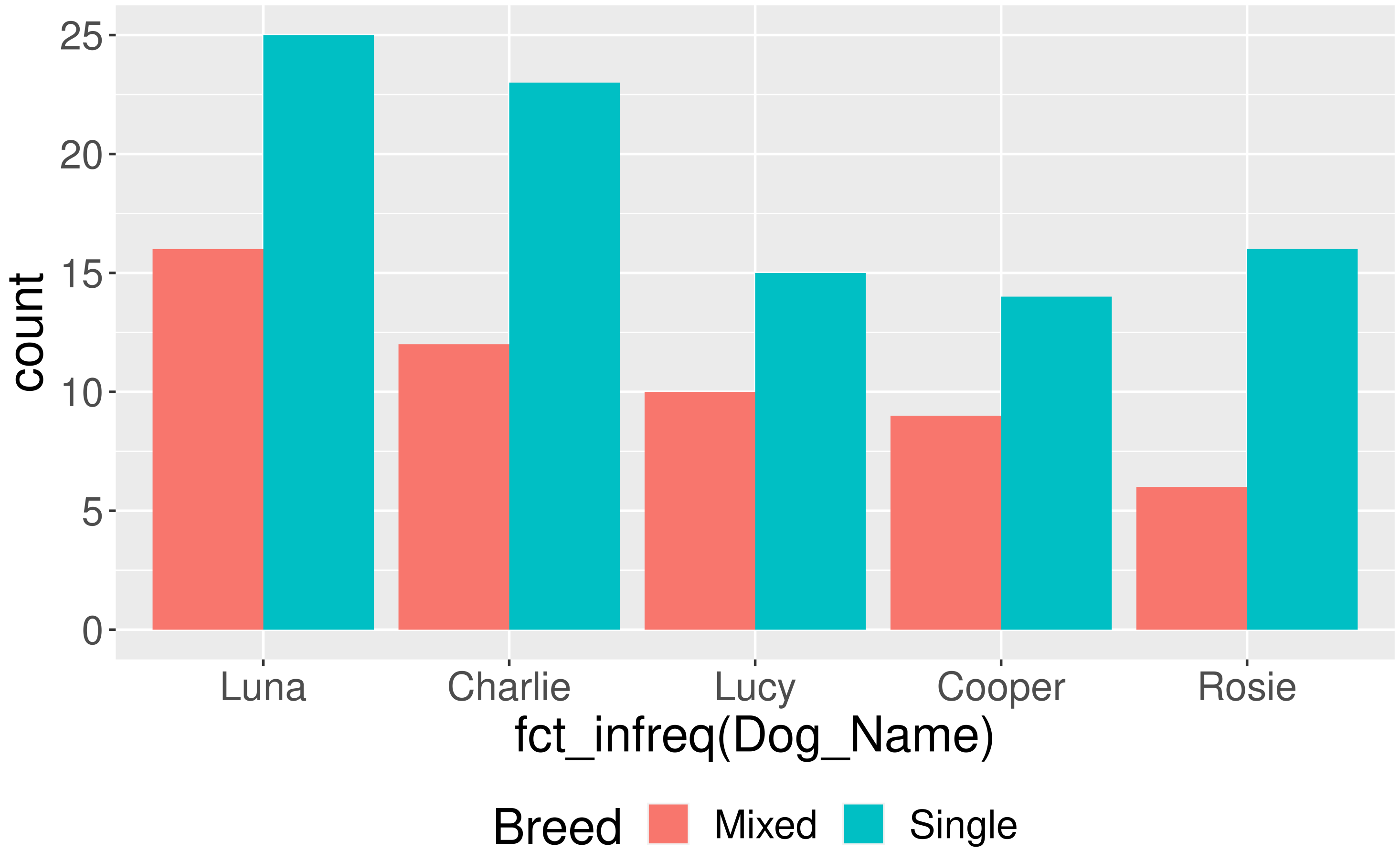
```r
# Create segmented barplot
ggplot(data = dogs_top5,
       mapping = aes(x = fct_infreq(Dog_Name),
                     fill = Breed)) +
  geom_bar() +
  theme(legend.pos = "bottom")
```



- Each bar is divided into the frequencies of the `fill` variable.
- Hard to make comparisons across categories.

Segmented Barplots

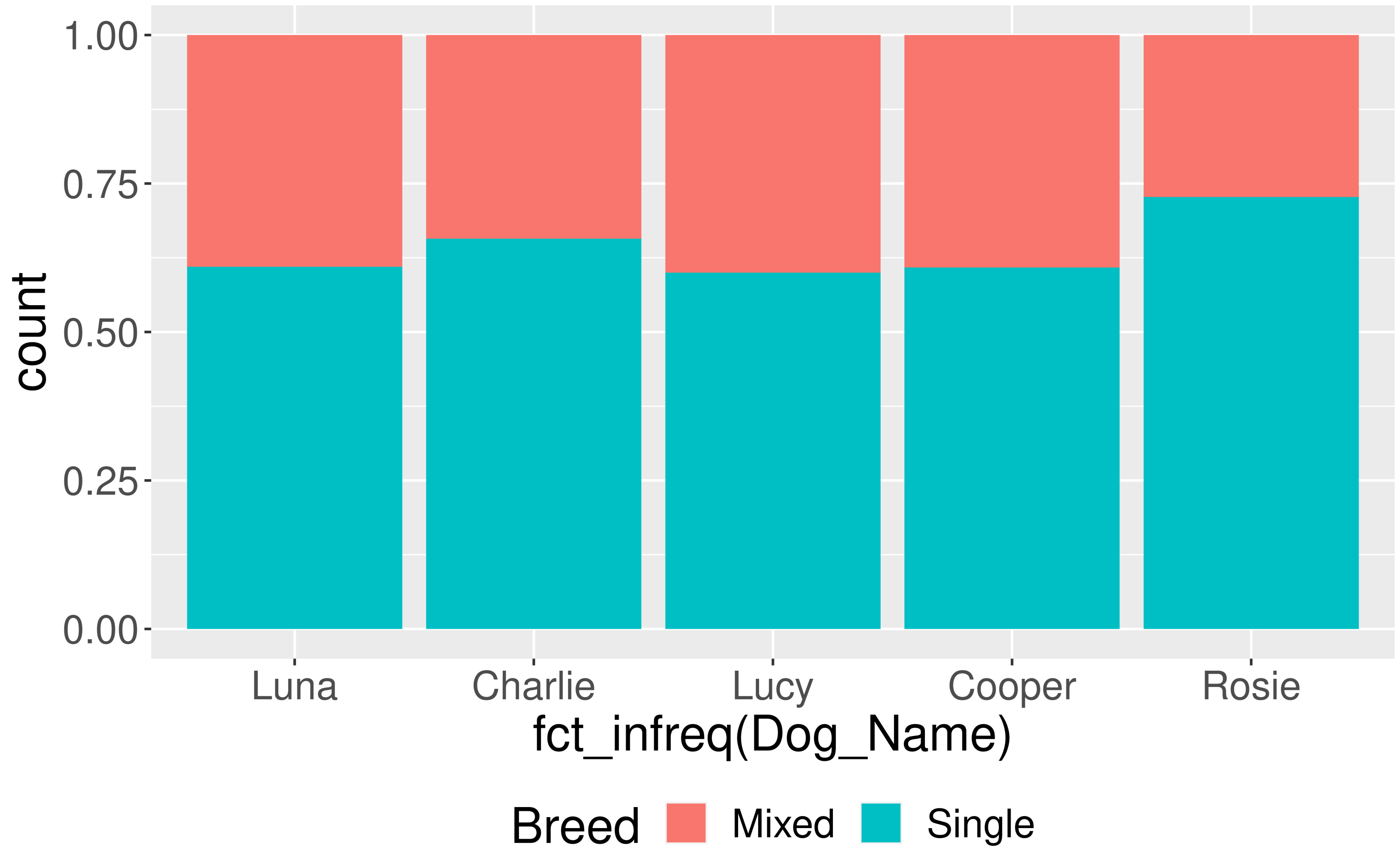
```r
# Create segmented barplot
ggplot(data = dogs_top5,
       mapping = aes(x = fct_infreq(Dog_Name),
                     fill = Breed)) +
  geom_bar(position = "dodge") +
  theme(legend.pos = "bottom")
```



- Can add the `position` argument into the `geom_bar()`.

Segmented Barplots

```r
# Create segmented barplot
ggplot(data = dogs_top5,
       mapping = aes(x = fct_infreq(Dog_Name),
                     fill = Breed)) +
  geom_bar(position = "fill") +
  theme(legend.pos = "bottom")
```



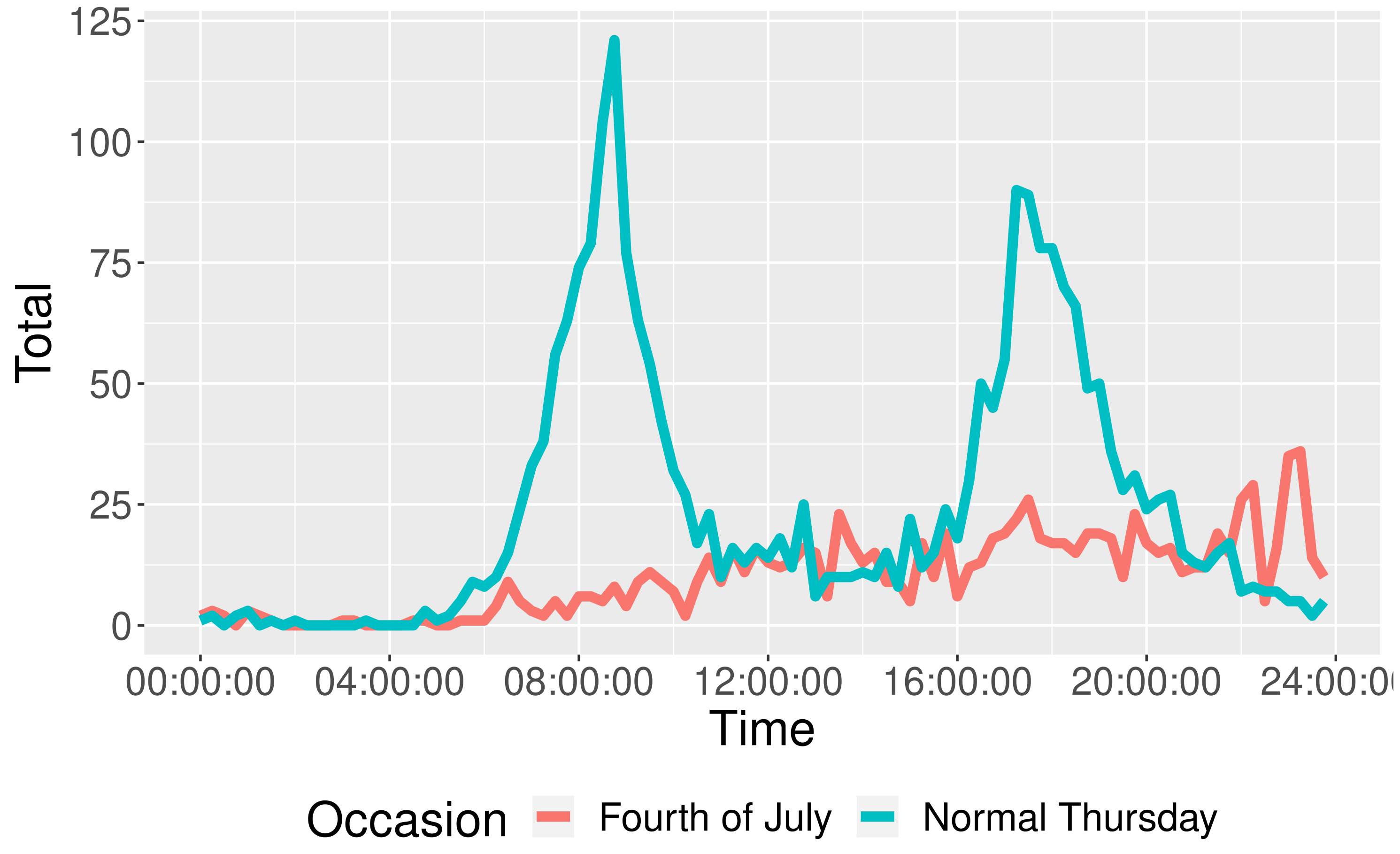
- Now each bar is divided into proportions based on the `fill` variable.

Adding More Variables

- Two main approaches:
  - Utilize other `aes`thetics of the `geom`
  - Facet: Create multiple plots across the categories of a categorical variable.

Utilize other aesthetics

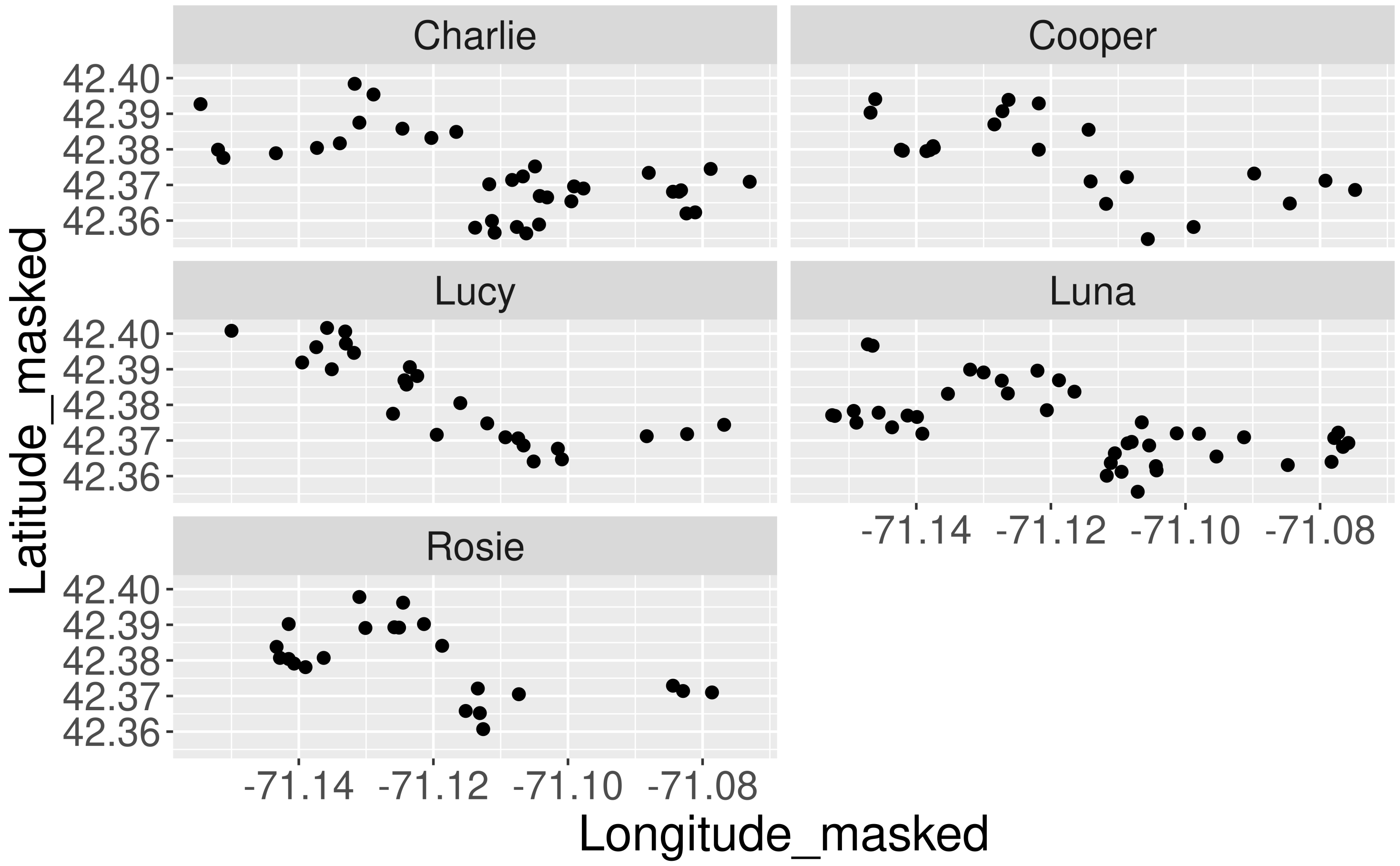
```r
ggplot(data = july_2019,
       mapping = aes(x = Time,
                     y = Total,
                     color = Occasion)) +
  geom_line(size = 2) +
  theme(legend.pos = "bottom")
```



- Already saw how to add a third variable to a line graph (and a scatterplot) via `color`.
  - Can also change size or type.

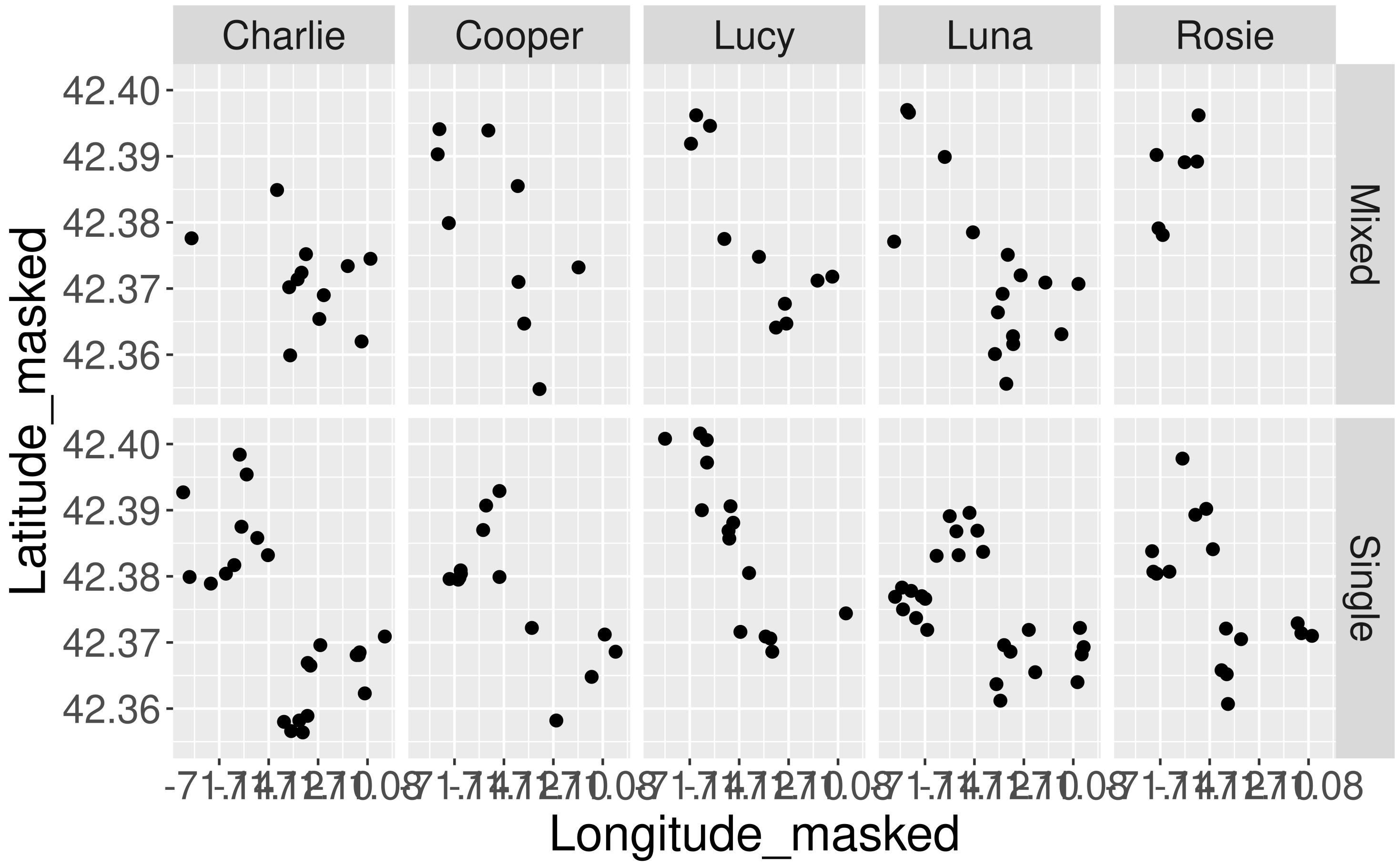
Facet

```r
ggplot(data = dogs_top5,
       mapping = aes(x = Longitude_masked,
                     y = Latitude_masked)) +
  geom_point(size = 2) +
  facet_wrap(~Dog_Name, ncol = 2)
```



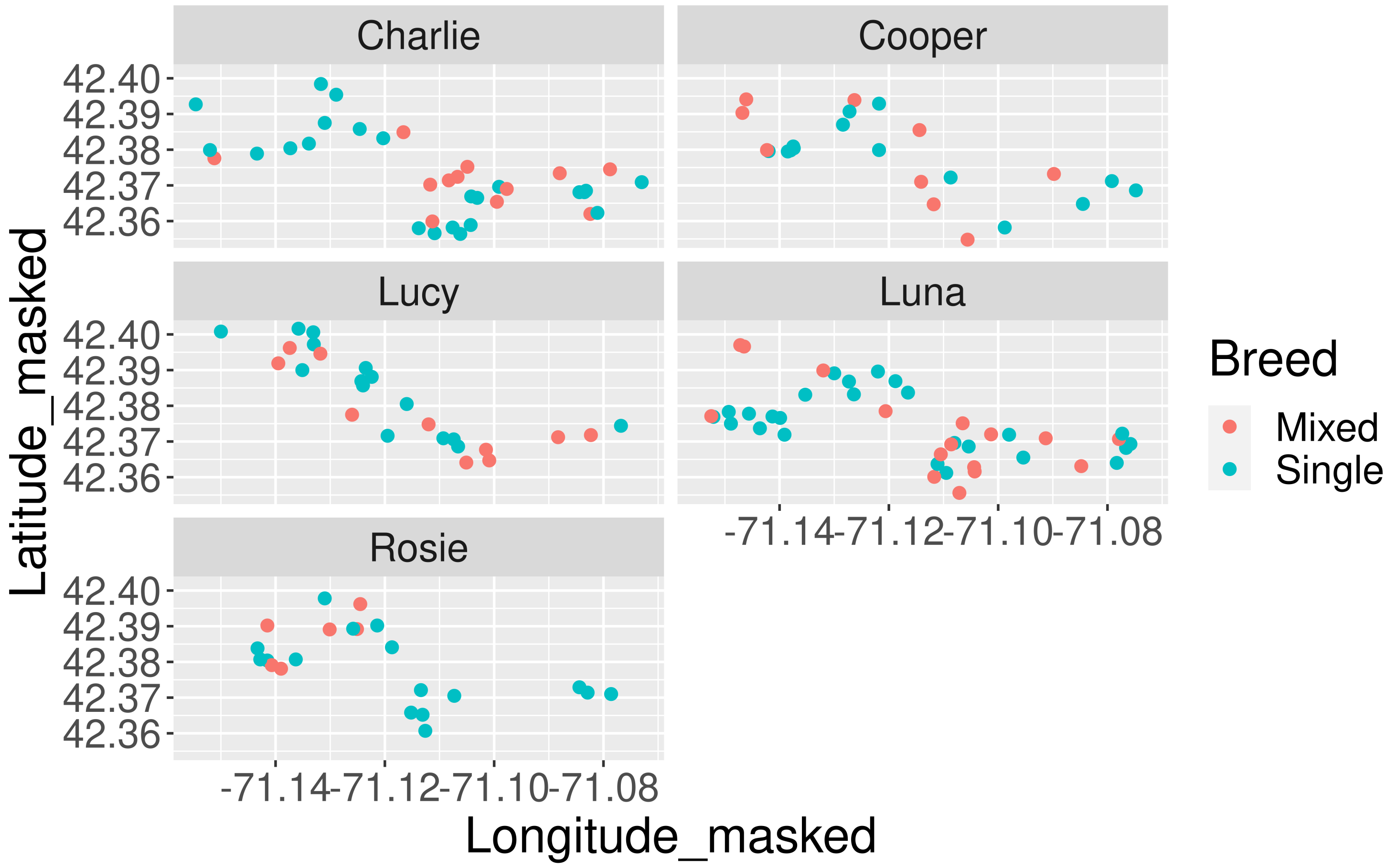
Facet

```r
ggplot(data = dogs_top5,
       mapping = aes(x = Longitude_masked,
                     y = Latitude_masked)) +
  geom_point(size = 2)  +
  facet_grid(Breed~Dog_Name)
```



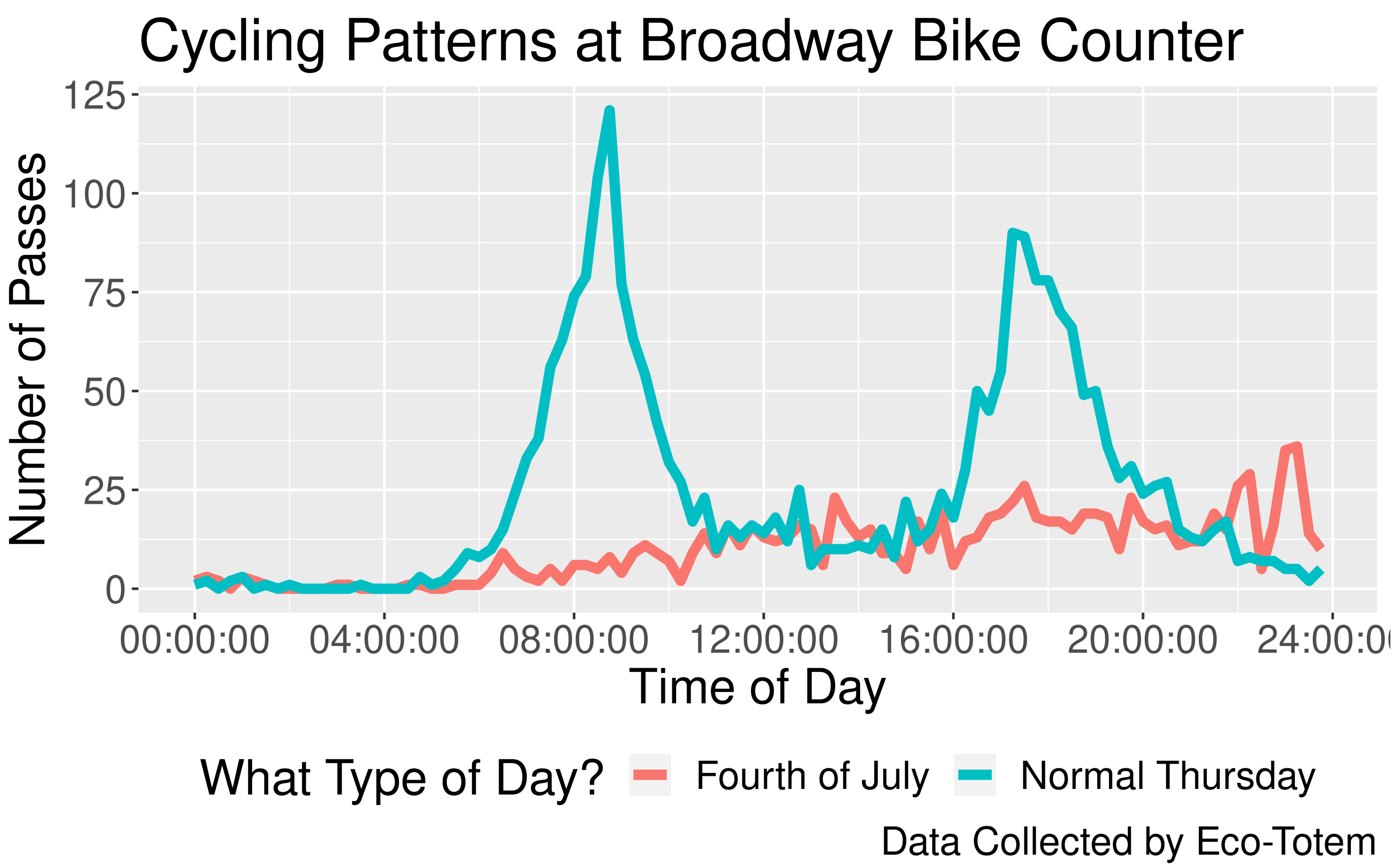
Consider Doing Both!

```r
ggplot(data = dogs_top5,
       mapping = aes(x = Longitude_masked,
                     y = Latitude_masked,
                     color = Breed)) +
  geom_point(size = 2) +
  facet_wrap(~Dog_Name, ncol = 2)
```



Adding Some Context

```
ggplot(data = july_2019,
       mapping = aes(x = Time,
                     y = Total,
                     color = Occasion)) +
  geom_line(size = 2) +
  theme(legend.pos = "bottom") +
  labs(x = "Time of Day",
       y = "Number of Passes",
       color = "What Type of Day?",
       caption = "Data Collected by Eco-Totem",
       title = "Cycling Patterns at Broadway Bike Count
```

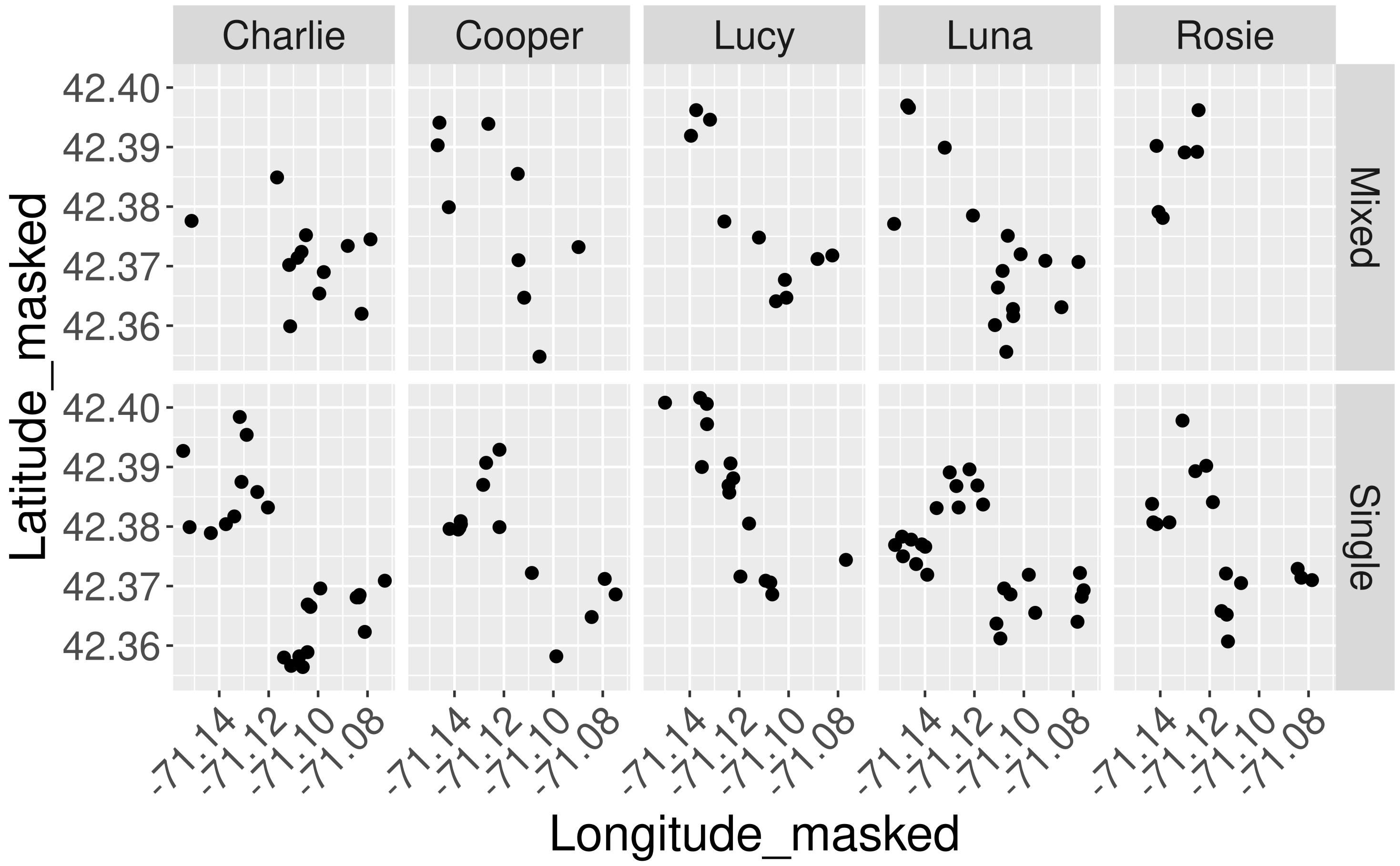


Customizing your `ggplot2` Plots

- There are so many ways you can customize the look of your `ggplot2` plots.
- Let’s look at some common changes:
  - Fussing with labels
  - Zooming in
  - Using multiple `geoms`
  - Color!
  - Themes

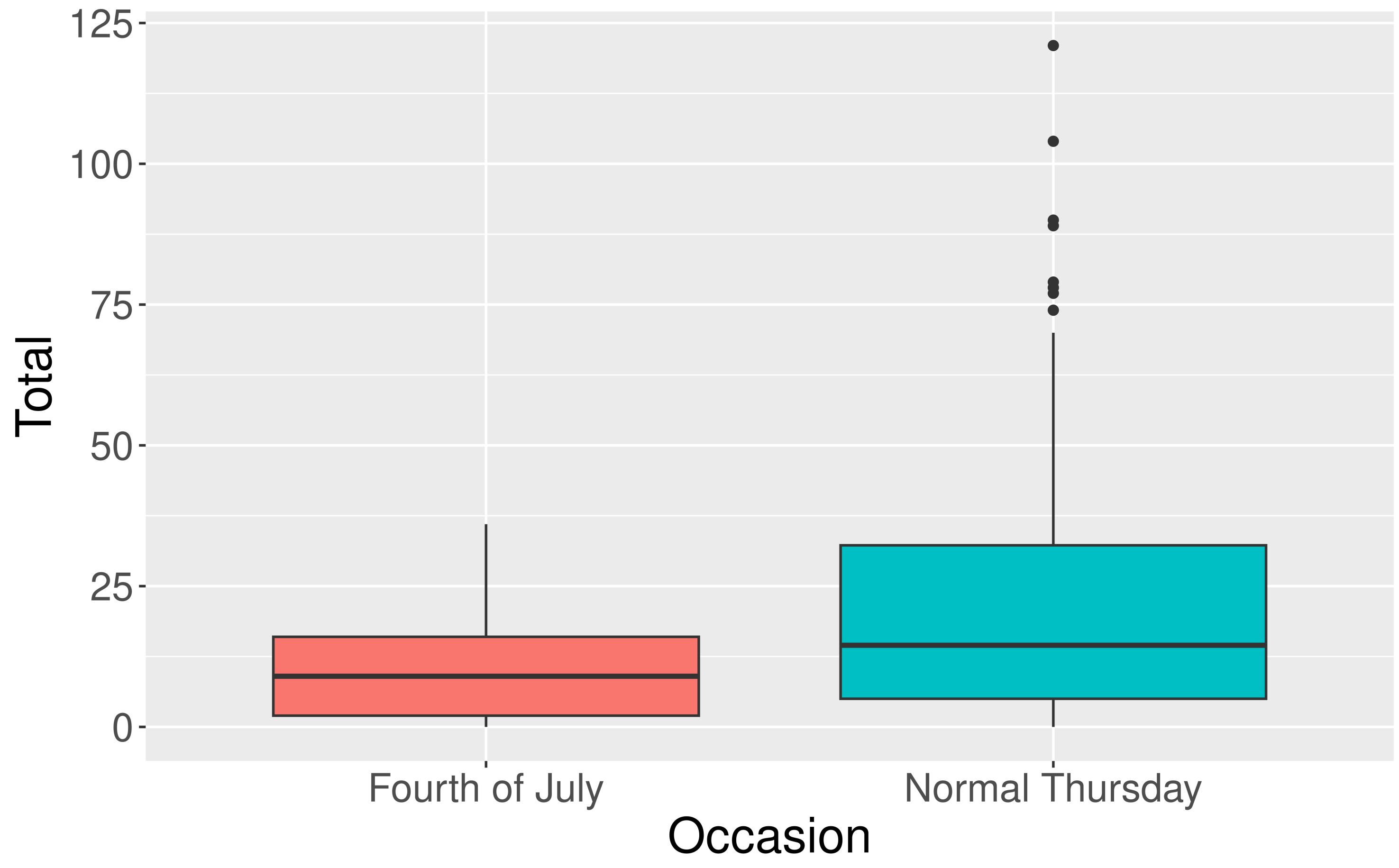
Fussing with Labels: Rotate

```r
ggplot(data = dogs_top5,
       mapping = aes(x = Longitude_masked,
                     y = Latitude_masked)) +
  geom_point(size = 2)  +
  facet_grid(Breed~Dog_Name) +
  theme(axis.text.x =
          element_text(angle = 45,
                       vjust = 1,
                       hjust = 1))
```



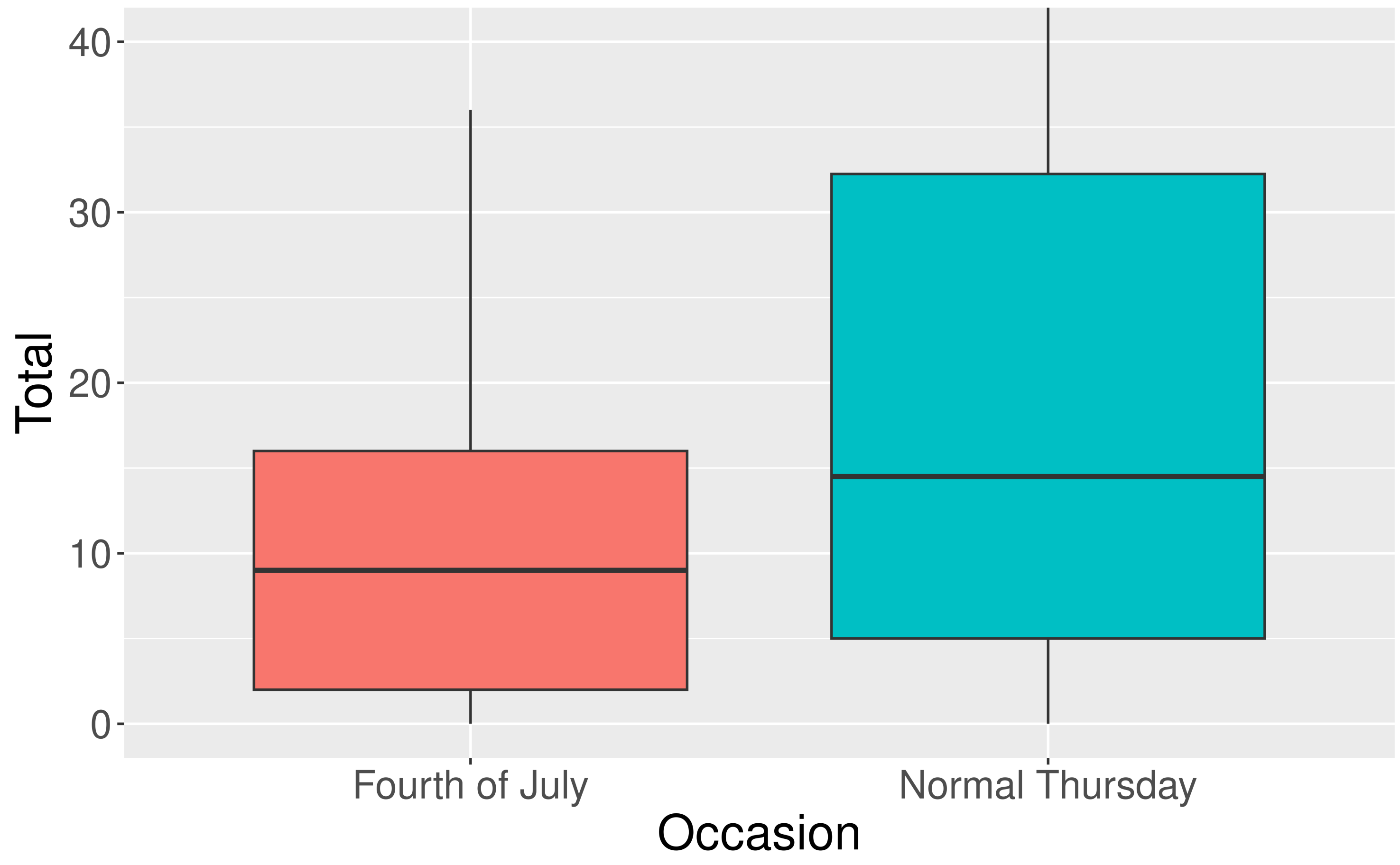
Zooming In

```r
ggplot(data = july_2019,
       mapping = aes(x = Occasion,
                     y = Total,
                     fill = Occasion)) +
  geom_boxplot() +
  guides(fill = "none")
```



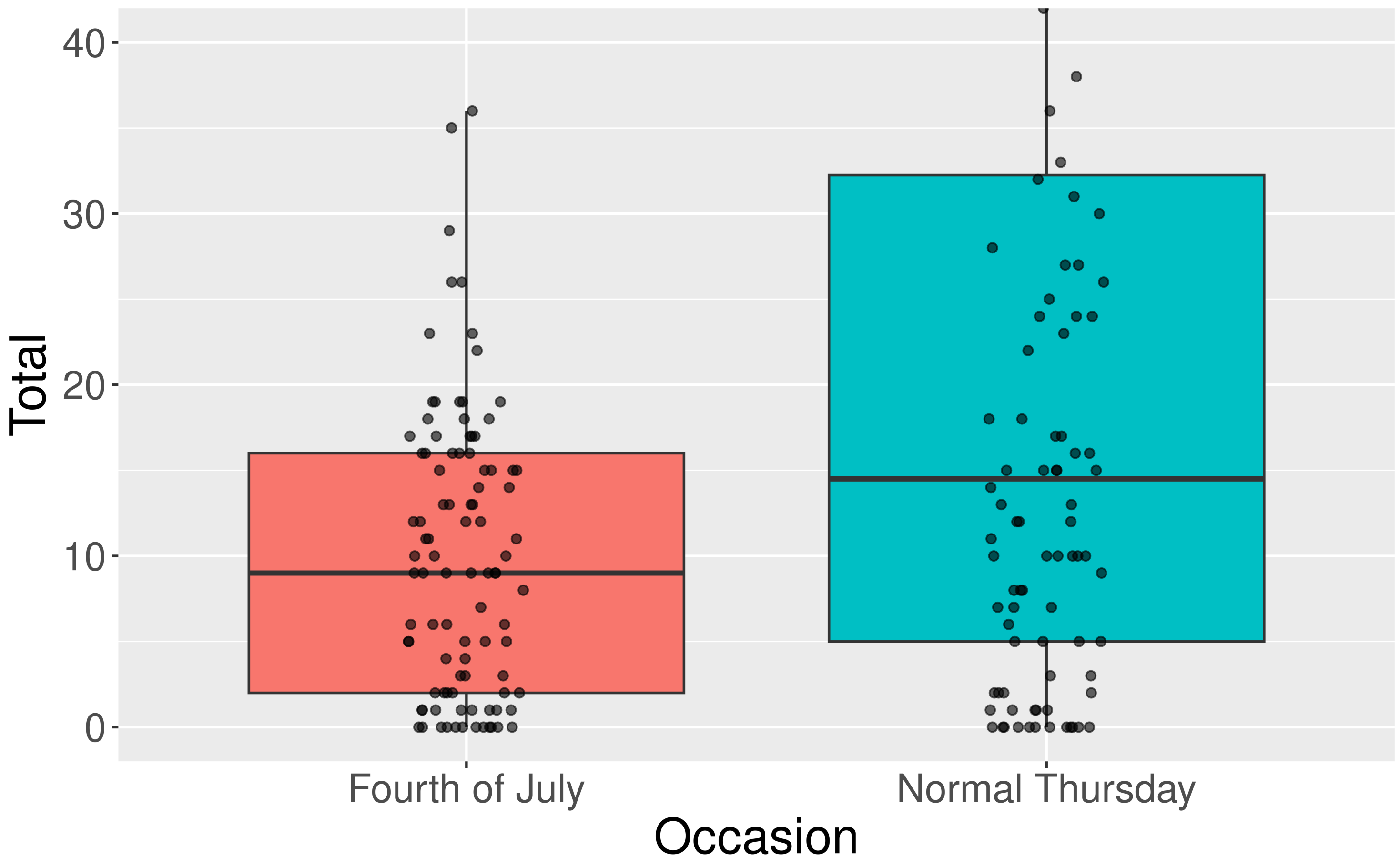
Zooming In

```r
ggplot(data = july_2019,
       mapping = aes(x = Occasion,
                     y = Total,
                     fill = Occasion)) +
  geom_boxplot() +
  guides(fill = "none") +
  coord_cartesian(ylim = c(0, 40))
```



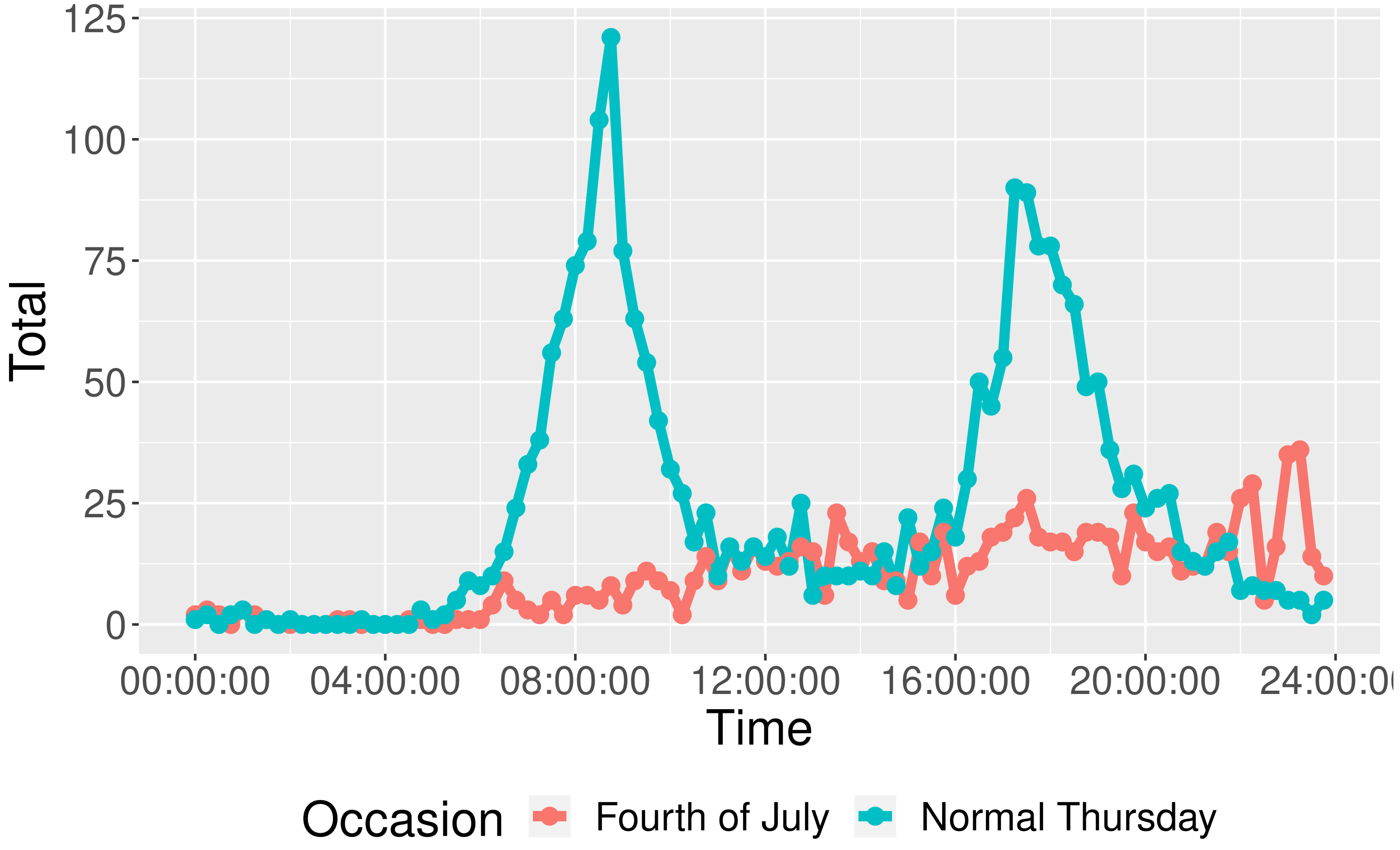
Multiple geoms

```r
ggplot(data = july_2019,
       mapping = aes(x = Occasion,
                     y = Total,
                     fill = Occasion)) +
  geom_boxplot() +
  guides(fill = "none") +
  coord_cartesian(ylim = c(0, 40)) +
  geom_jitter(width = .1,
              height = 0,
              alpha = 0.6)
```



Multiple geoms

```r
ggplot(data = july_2019,
       mapping = aes(x = Time,
                     y = Total,
                     color = Occasion)) +
  geom_line(size = 2) +
  theme(legend.pos = "bottom") +
  geom_point(size = 3)
```



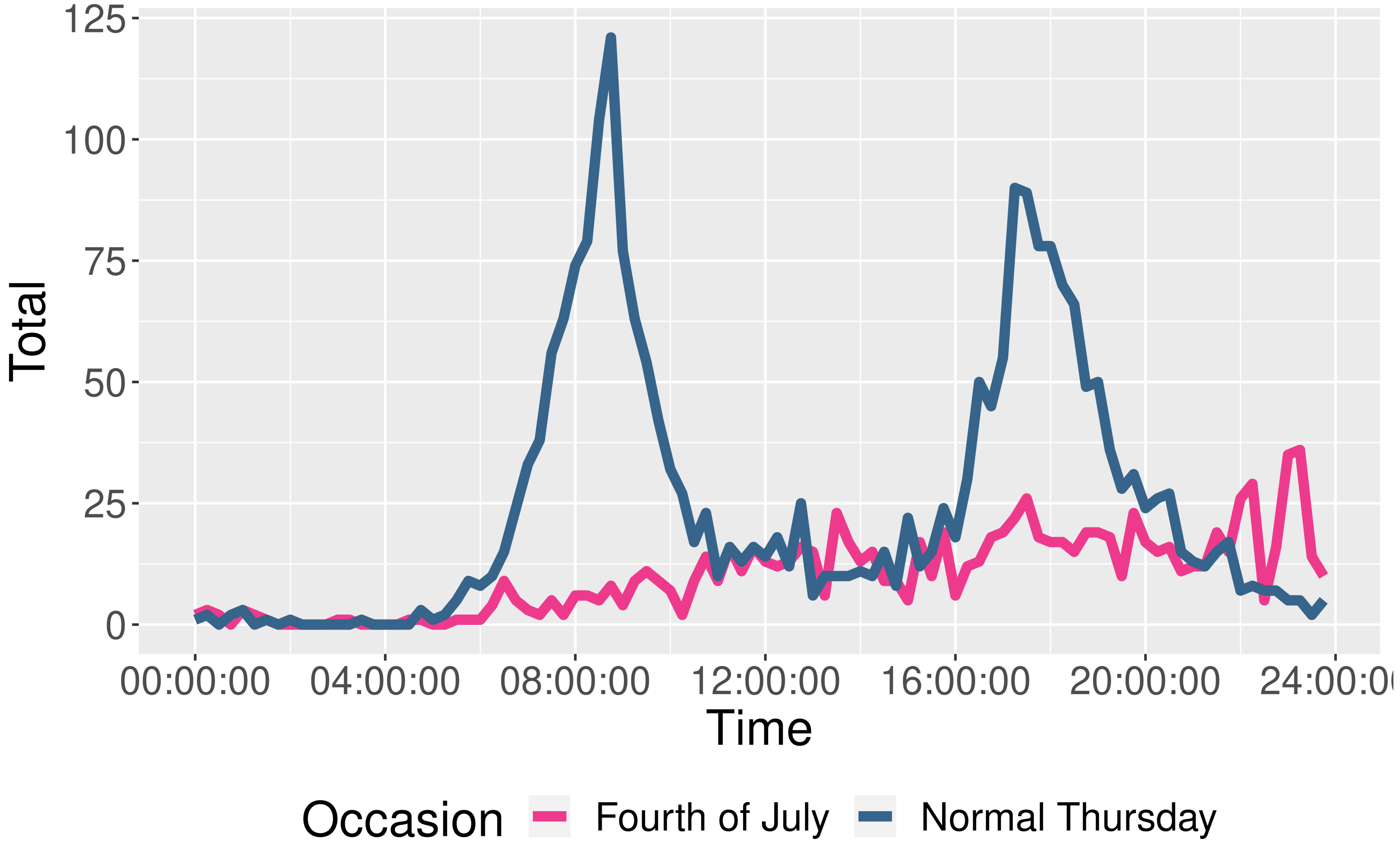
Change the Color

```r
colors()
```

```
 [1] "white"                "aliceblue"            "antiquewhite"
 [4] "antiquewhite1"        "antiquewhite2"        "antiquewhite3"
 [7] "antiquewhite4"        "aquamarine"           "aquamarine1"
[10] "aquamarine2"          "aquamarine3"          "aquamarine4"
[13] "azure"                "azure1"               "azure2"
[16] "azure3"               "azure4"               "beige"
[19] "bisque"               "bisque1"              "bisque2"
[22] "bisque3"              "bisque4"              "black"
[25] "blanchedalmond"       "blue"                 "blue1"
[28] "blue2"                "blue3"                "blue4"
[31] "blueviolet"           "brown"                "brown1"
[34] "brown2"               "brown3"               "brown4"
[37] "burlywood"            "burlywood1"           "burlywood2"
[40] "burlywood3"           "burlywood4"           "cadetblue"
[43] "cadetblue1"           "cadetblue2"           "cadetblue3"
```

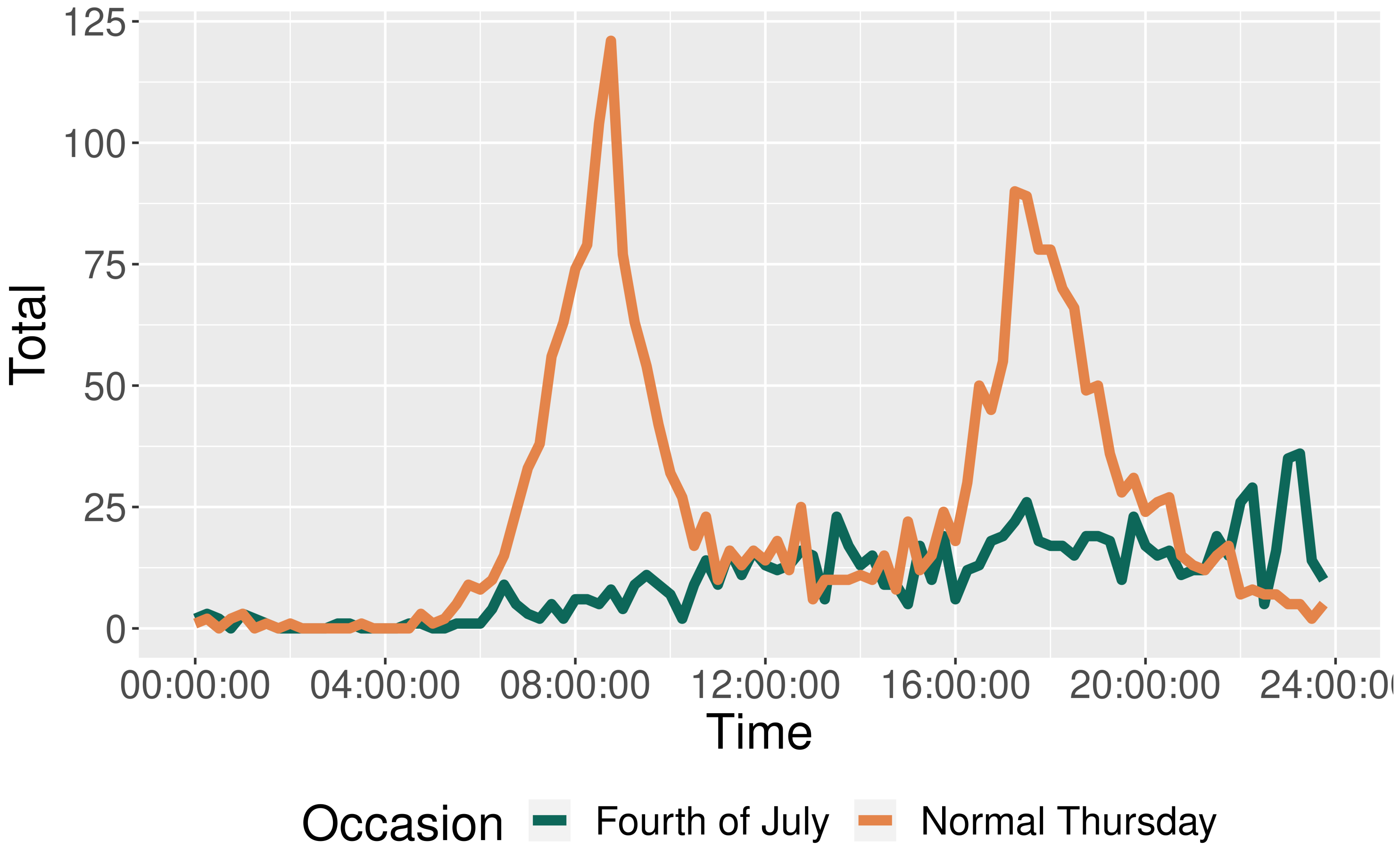
Change the Color

```r
ggplot(data = july_2019,
       mapping = aes(x = Time,
                     y = Total,
                     color = Occasion)) +
  geom_line(size = 2) +
  theme(legend.pos = "bottom") +
  scale_color_manual(values = c("violetred2",
                                "steelblue4"))
```



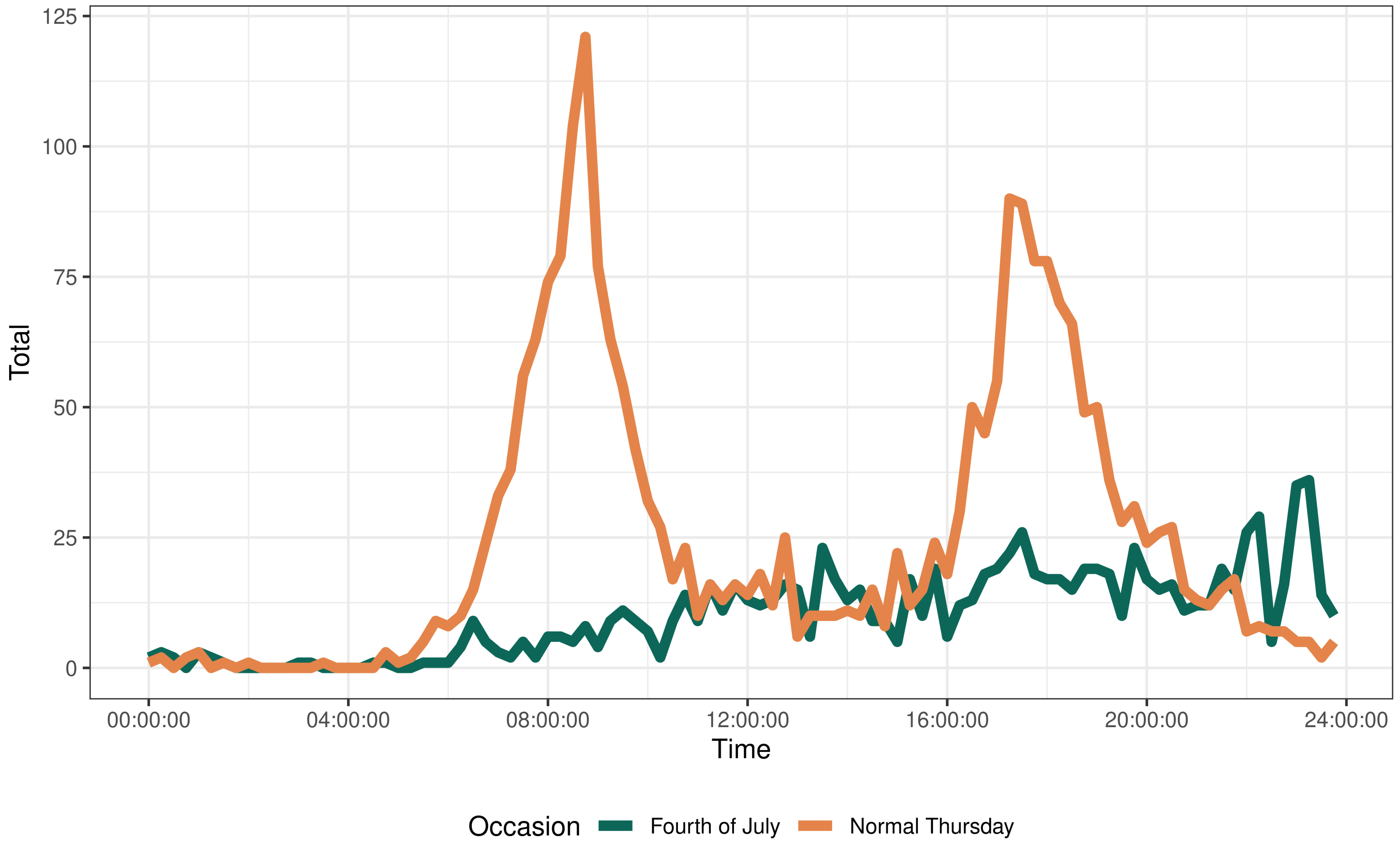
Change the Color

```r
ggplot(data = july_2019,
       mapping = aes(x = Time,
                     y = Total,
                     color = Occasion)) +
  geom_line(size = 2) +
  theme(legend.pos = "bottom") +
  scale_color_manual(values = c("#0D6759",
                                "#E4844A"))
```



Use a Different Theme

```r
ggplot(data = july_2019,
       mapping = aes(x = Time,
                     y = Total,
                     color = Occasion)) +
  geom_line(size = 2) +
  scale_color_manual(values = c("#0D6759",
                                "#E4844A")) +
  theme_bw() +
  theme(legend.pos = "bottom")
```



What `ggplot2` questions do we have?

Reminders

- With COVID working its way through campus right now, make sure to check the Sections spreadsheet and the Office hours spreadsheet for updates!
- Grab a **postcard** and/or a **stamp** from SC 316 if you lost yours.
  - We also have markers, colored pencils, and crayons!
- Don’t forget that P-Set 1 due on Tuesday by 5pm in Gradescope.
- Come by office hours with any questions.