Okay, so now we're going to start with a simple bar plot of the MIT international student data.

So first, let's load the ggplot library, ggplot2, and load the data frame.

So intl = read.csv("intl.csv").

Now, the structure of this date frame is very simple.

There are two columns, two variables.

The first one, the region, and the second one is the percentage of international students who came from that region.

So making a bar plot from this data isn't too hard.

We start off with a ggplot command, of course, the first argument being the data frame.

The aesthetic in this case is to have Region on the x-axis, and on the y-axis, to have the percentage of international students.

Now, the geometry we're going to use is, as you might guess, bar, geom_bar.

We have to pass one argument to this geom_bar, and it's called stat = "identity" .

I'm going to come back and explain what that means.

I also want to label my bars with the value, so it's easy to read in closer detail.

So I'm going to use geom_text to do that.

And the aesthetic of our text is simply to have the value of a label, the text of a label, to be the value of our percentages.

Let's look at that.

So yes, we have a bar for each region.

The values are between zero and one, which looks kind of strange.

The labels are actually lying over the top of the columns, which isn't very nice, and the regions aren't really ordered in any way that's useful.

They're actually ordered in alphabetical order, but I think it would be much more interesting to have them in descending order.

So we're going to work on this.

First of all, though, what is this stat = "identity"?

Well, it's pretty simple.

Geometry bar has multiple modes of operation.

And stat = "identity" says, use the value of the y variable as is, which is what we want.

The height of the bar is the value of the y variable.

Now, there are other modes, including one that counts the number of rows for each value of x, and plots that instead.

So you can look at the documentation for ggplot to see the different options and how they work.

But stat = "identity" is what we want right now.

Now, the x-axis is out of order.

And the reason for this is that ggplot defaults to alphabetical order for the x-axis.

What we need to do is make Region an ordered factor instead of an unordered factor.

We can do this with the reorder command and the transform command.

So let's write this out.

So we're going to transform the international data frame.

And what we're going to do is say, Region, it's going to be a reordering of itself, based on decreasing order of PercentOfIntl.

So if we look at the structure of the data frame now, we see there's something going on in the Region column that wasn't going before.

And that's that ordering.

So you might have also noticed that I put a negative sign in front of PercentOfIntl.

So that negative sign means decreasing order.

If we had left that out, it would have actually ordered them in increasing order.

So unknown or stateless would have been first, and Oceania would have been second, and so on.

So that's one thing fixed.

Another thing we didn't like was that the numbers were between zero and one, which looks a little bit messy.

So let's just simply multiply all the values by 100.

So intl$PercentOfIntl = intl$PercentOfIntl*100.

And now the other things we have to fix, like the text overlying and the x-axis being all bunched up like that, we're going to do that in a new ggplot command.

So I'm going to break it across multiple lines.

So we start up with the ggplot command, as we did before, actually identical to what we had before.

So the aesthetic is x-axis is the region, and the y-axis is the percentage of international students.

We break it into multiple lines, so put the plus at the end there, and press Shift Enter.

We're going to do a bar plot.

The stat = "identity", as before.

And this time though, we're going to manually specify a fill.

I'm going to say "dark blue".

I quite like how that looks.

Now, we need the text, and the aesthetic of that is to have the label equal the value of the column.

I'm going to add one more thing to this.

I'm going to say vjust = -0.4.

And what this does is, it moves the labels up a little bit and off the top of the bars.

You can play with that.

So a positive value will move it down, and a negative value will move it up.

Next, I'm going to set the y-axis label to be something a bit more sensible-- so "Percent of International Students".

And finally, I'd like to fix up that x-axis.

So I want to get rid of the word "Region," because it's pretty obvious these are regions.

And I also want to rotate the text at a bit of an angle, so you can read it all on a plot like this.

That's done with the theme command.

So the theming we're going to do is we're going to say the axis title, the x-axis, should be blank.

And the axis text on the x-axis should be rotated, so it's a text element that's angle is 45.

And I'll move it sideways just a little bit-- hjust = 1.

And there we go.

So we've got our labels vjust-ed above the columns.

The bars themselves are dark blue.

The numbers are now between 0 and 100, instead of zero and one.

We can read all the text labels.

And it's generally a lot more readable than the pie plot or our original ggplot, at that.

Let's go back to the slides now and talk about what we've just done.