All right.

So we've already gotten pretty far. But the next thing that I want to create are these little boxes. These boxes that contain an icon and also a phone number or an email address, whatever it is that you want to show.

Now you can use an image asset as an icon. But it's nowhere near as adaptable as simply using an icon widget.

And I want to show you how this works.

If we collapse both of our text widgets and our circle avatar, then we can see that in our column, we currently have three things, our circle avatar, our text which is our name, and finally, the job title that we have here.

So I'm gonna go ahead and add another child and this is gonna be a container. Now inside this container, the only child that it's gonna have is a row.

And very often when you're creating user interfaces in Flutter, you'll find yourself embedding rows inside columns or columns inside rows.

And this gives you that kind of grid layout pattern that makes your layout really flexible.

We're going to have a fourth child which is going to be a row, and this row is going to have a whole bunch of children including an icon. Now an icon is a class that is drawn instead of being shown.

So for an image it's simply displayed as is.

But for an icon, it's actually drawing all of the sides and the edges onto the screen.

What this allows you to do, is you can on the fly for example, change the color of the icon or the size of the icon.

And it's so much more flexible than simply just using an image.

And by using the material.dart package from Flutter, you already have access to a whole bunch of icons that Google made. So, if you take a look at the documentation for the icons class, icons with an s, you can see that you have all of these icons at your fingertips. And they come from the material design icons.

So these are icons that designers at Google actually created, so you can have a consistent theme across apps that are built with the material design look. The way that I prefer to browse icons is actually using a tool called materialpalette.com.

And not only is this really good for coming up with really intelligent beautiful color palettes. So for example on the main page, if you select two colors that you quite like, for example teal and yellow, then it'll automatically generate you a eight-color color palette even complete with an accent tone. And you can see it looks really really nice straight up from the get go. And it also has the colors that we've been using so far and gives you their names all in one place.

And if you click on them, it'll show you all the various shades that you can tap into.

There's also a third tab which are the icons. And it's a really easily searchable place where you can say let's look for an add icon, and it shows you here are all the ones that you could possibly use.

And when you find one of them, you can see it has a name.

Right.

This one's called add\_shopping.cart. So I can simply go into my Flutter app and write icons, which is the name of the class that supports material design icons.

.add\_shopping\_cart. And now, almost immediately, you can see that I can add that icon that I found on here to my app. Which is brilliant, if you don't want to spend a lot of time finding or creating or paying for these design resources.

Now I mentioned that icons are better than images because they're drawn on the fly.

So let me show you what you can do with these icons that you can't do with images.

I can for example turn that icon into a much larger icon by simply changing its size property.

So let's make it 100, and you can see that almost immediately, my icon becomes super large and the lines get thicker and it scales until infinity.

So if I change this to 1000, you'll see that it won't pixelate because it's actually just drawn.

All that happens, is that it goes off the screen, and you get these yellow bars as a warning to tell you that there's parts of your content that's not visible to the user. It's completely off the screen but you can see that this icon doesn't become pixelated, whereas if I did that to my image up here.

So let's say I make that circle avatar 1000 in its radius. Then you can see that my picture becomes a little bit pixelated. And let's go a little bit more extreme, let's add another zero to that, and now it's completely not interpretable.

I can't really tell what that is anymore even though I know it's her.

This is the reason why icons are much better because they're vectors, and you can even change their color on the fly.

Let's change the color of my icon to that teal color that we've been using. Just save and you can see it now blurs into the background. But if I change that into a 900 shade, and it save, then you'll see it become a much darker teal.

And now it's completely visible.

So you can change the color of it.

You can change the size of it.

You can do all sorts of things with these icons that you can't do with images.

And this is why they're really really useful inside your app.

All right.

So the icon that we actually want here is the one called phone.

And this one gives us that little phone icon that we can put next to some piece of text.

So in addition to the icon, I'm also going to have some text and this is simply going to be a phone number.

So +44 123 456 789 for example.

And then I'm going to style up my text because at the moment it looks a bit boring.

So I'm going to give it a text style. And this is going to have a color which is going to be a dark teal color.

So colors.teal.shade900 and then it's going to have a font family of the same one that we've been using which is Source Sans Pro. And we can change the font size to 20 points, and let's hit save and see what that looks like.

All right.

So that looks pretty nice but it'd be nice if there was a little bit of space between the icon and the text right.

Well we know that we are inside a row, and the way that we can easily add space between our children in our rows, is by adding a sized box.

So we're going to add a sized box and we're going to give it some dimension in the horizontal axis.

Right.

Because we're in a row, so it's going to go from left to right.

So we'll give it a width, and let's give it a width of maybe 10 points.

And that should be enough to separate those out a little bit.

Make it look just that bit nicer.

So finally I'm going to give my container a color just so that it will show up as a separate piece of information, from the rest of my widgets.

So let's give this a color of white, and let's hit save, and now I have to change my icon back into teal so that it's visible. And I can make it come away from the edges of the screen a little bit by adding margin.

So a lot of the widgets inside Flutter will have margin or padding as properties that you can change.

And they all work exactly the same way as we saw earlier with the columns and the rows.

We use edgeinsets and I'm going to add edgeinsets only that are different for the top and bottom and also left and right.

So I'm going to use some metric. And I'm going to say that the vertical sides, so the top and bottom, are going to have 10 points of margin and the horizontal is going to have 25 points of margin.

Now you can see that our container comes away a little bit from the edges, and it looks just a little bit better. And our container also has padding.

So let's add maybe 10 pixels of padding to all four sides. And now we have our little card.

Right.

As a challenge I want you to create the other container. The one where you have your email address and an email icon.

So I want to challenge you to use the Flutter documentation, and also what you learned in this lesson to create that second card. Pause the video, and give that a go now. OK, so we know that we need another container.

Right.

It's pretty much the same as this one.

So let's find out where our container ends.

It's right about here.

And let's go ahead and collapse it. And then we're going to add another container underneath it.

And this one is also going to have the same amount of padding, so that was edgeinsets all with 10 points.

And then it's going to have a margin of symmetric margin with 10 on the vertical, and 25 on the horizontal.

And finally it's going to have some content and the content is going to be under the child property and it's going to be a row that we want to create.

The row is going to have some children. And these children include, one a icon that comes from the icons class. And you can either scroll through this to find an icon that looks right.

But in this case it's going to be email. Or you can take a look at the websites that we spoke about earlier on.

Now that I have my e-mail icon in there, I'm going to change the styling of that icon to make it look a bit more like the other one.

So I have to change its color to teal. And then I get to add my text which can really say anything you want it to.

But in my case I'm going to add maybe an email. Let's say angela@email.com. And then I'm going to change its style to have a text style, and maybe we'll go for a font size of 20 to match with the other one, a color of teal 900 and the font family is of course also going to be Source Sans Pro. Let's hit save and let's see if we get what we wanted.

OK.

So we can see that the icon has disappeared.

So it's the same color as the background.

We've got our text formatted in the same way as the previous one, and all we have to do now is to give our container a white background and it should show up underneath the other one and show us what else is missing.

So the only thing that looks odd to me is just a little bit of space between the icon and the text.

So to do that we have to add our Sized Box with a width, and that was going to be 10.

And now if we hit save, then it's starting to look pretty good. In the next lesson, we're going to finalize this app, and we're gonna make these in those actions a little bit nicer looking using a Flutter widget called card.

So for all of that and more, I'll see on the next lesson.