Flexbox in Practice

Hello and welcome to the lesson about working with flexbox in Divi. So, in this video, I would like to walk you through a few practical ways you can use the CSS flexbox, when working with the Divi Builder will cover aligning modules, vertically switching, the order of columns and using flexbox, to create any number of columns inside at the row. So, let's get to work.

Use Case 1 - Flexbox Overview

I have a sample layout here with a few problems that we will be able to fix with flexbox. So, in this first section sometimes you have an image that you want to position at the bottom of the section. So that it - you know, sits there regardless of the amount of content you have on the other side. So I have a simple image module and a text module here and now if I remove the padding from the section, and also from the row. It still may not be enough to kind of position that at the bottom. And here we don't have enough space for the content. So we do want to have that space here but we want to make sure that this image is at the bottom. So to align it correctly, I would go to my module, text module setting, and add some spacing for this module. And now to align this image at the bottom we'll use flexbox, but to make sure we have enough room here, we need to equalize the column heights in the row settings, we will be targeting column as the parent container to be sure that this column has equal height as this one. We have to enable that option here. And now, in the First column settings - Advanced - custom CSS, I can simply define display flex. And align-items: flex-end. Simple as that. So now it can cause troubles, especially when, let me save that, especially on smaller screen sizes. Let's say that would be my image here at the bottom, that's fine. But now it kind of floats at the top here. So on mobile, I would like to reverse the order of these columns. I would like to display my image at the bottom here. So it's still, it's kind of fixed to that next section. And I can also do that easily with flexbox. So again in the row and this time we'll be targeting a row to change the order of columns. So in the Advanced - custom CSS main element but only on tablet. I can say display-flex, and flex-direction column reverse. And this will change the order of these two columns on mobile and let me preview that on the front end. So now if I refresh the page and my tablet is displaying the columns in a reverse order. Now, there is a problem here with the spacing and that is caused by the bottom margin, which is added to the first column for tablets inside Divi CSS. So we want to make sure that this column has no margin at the bottom when we are using this method. So, here in the column settings in the design spacing, we don't have any margin here, so we can only do that with CSS, custom CSS. And then for the tablet, we want to copy that CSS, or we can actually, add that for all devices. Like that. And now if I save and refresh, that should fix the alignment here. So let's look at the next section.

Use Case 2 - Centering Content

Here is another common situation where we have an image and text or any other content here. And we want to align this content in the center, regardless of the size of the image. And we can again do that very easily with a similar property. So, inside the row settings, we have to make sure that column uses the same height. So equalize column heights. That's the first step and then again in the first column, we want to align the content of that column with flexbox. So Advanced - custom CSS - Main Element - display: flex, and align-items: center. Simple, as that.

Use Case 3 - Aligning Multiples

Okay, a different situation. I have three cards, three blurbs, and a button. And I used a column to display the background. So, it's not just a blurb module, but I'm using multiple modules and I'm using a column to display the background color, because when I use the design as a row setting and equalize column heights that will give me this nice equal height cards layout. But the problem we may have here is that we may want to align these buttons in the same position. We could do that with position: absolute, but we can also do that with flexbox very easily. So, in the column settings, let's go to the first column, Advanced - custom CSS, and again, we define display: flex. Which by default uses the direction row. So you want to change it to the flex-direction column. And we want to justify-content. and let's use space between because that leaves no space at the top and at the bottom here and that is exactly what we want and now we can extend this CSS to all columns throughout this section. Let's have a look, lovely. So even if I have more content or less, my buttons are nicely aligned at the bottom, simple as that.

Use Case 4 - Exceeding Divi Limits

So another situation where the simplest solution is to use flexbox, is when we want to display more than the default six columns. So at most with Divi, we have the six column layout, but what if I want to display eight or ten tiny logo images in a row, so I can do that again very easily with flexbox. And this time also I'm using a single column structure here and so my column would be my parent container. So inside the column settings - Advanced - custom CSS and again, display: flex, align-items: flex-start, that places them to the top. If you have different images that you could use align item center and we want to add a flex-wrap property to wrap because when there isn't enough room for our small images, they will wrap on to the second line, and then with the justify-content, we can decide if you want the modules to be displayed to the side like this, with the default flex-start. Or maybe if we want to space it evenly, which would kind of add space around. And again, when there are less items in a row, it would also align them evenly. So that's one approach. And now if I save, exit the builder. I can check how these logos look on smaller screen sizes. So with flex-wrap wrap, we can make sure that they fit on any screen size, but if the screen size is wide enough, it can display more than six in a row.

So I hope these examples are helpful. I'm sure that once you get familiar with flexbox, you will find lots and lots of use cases and places where it's helpful to align different website elements. Thank you for watching and see you in the next video.