**Traditional string output**

Selecting transcript lines in this section will navigate to timestamp in the video

- [Instructor] Template Literals are great for outputting HTML and other content mixed with JavaScript. However, this is a very new technology and chances are, you'll come across existing code and tutorials where Template Literals are never used, instead you'll see standard strings used. This is the classic way of outputting string content using plain old strings. It's a bit more convoluted, but it is and will be used. So it's worth knowing about. To show you this, I need to simplify my example, otherwise we'll just end up with way too much code. So I'm going to delete the entire Template Literal. And instead set content equal to a string that holds an h1 and name goes here, and then closes the h1. If I save this, we'll just get an h1 saying name goes here on the browser, because we are still doing the same thing we're still taking the value of this, injecting it into the inner HTML of the body and the document. So far everything is the same, we just have a shorter string to display. But now I want to grab this name property from the everyday backpack object and display its value in here. In the Template Literal I did that by typing in dollar sign, curly bracket, everyday pack, and then the property name. But you can already see that this is not working the same way, right? VS Code is not trying to help me here telling me, oh, yeah, you want the object and do you want this property. And if I save this, I just get that template tag out as a string in my browser, absolutely not what I want, right? That's because we are now outputting just a string and strings do not accept any type of template tags to just output whatever you put inside the quotation marks. So to get the content we want, that means we need to break this string into pieces and use string concatenation. So I want the string (h1), then I'll add a quotation mark to break out of the string, then I'll use a + symbol to say. Also I want to append on to the string, the value of everyday pack Dot name, that gives me the name property. Then, + again because I also want to append the string for the end h1 tag. So now if I save this, we actually get the output we want. And I can now change the value up here to something else and you'll see it changes so everything is back the way it was. But you can see this is far more complex and also far more full of clutter. And most importantly we can no longer operate with a proper template. Because now we're operating inside string territory and strings do not respect White space or anything else. So like I said, this is not the preferred way of doing things anymore. We have Template Literals for a reason. And it's specifically to solve this problem of string concatenation. But that said, you will encounter this formatting all the time when you work with JavaScript. And now you know what it is. Anytime you see a string followed by a + symbol, and then some value, you know this is string concatenation, and it's trying to output a mix of HTML and JavaScript expressions at the same time. That also means this is probably an opportunity to upgrade the code to use Template Literals instead.