00:00 Okay, let's do some dynamic content, through JavaScript onto our HTML page. And Normally our data would be coming in maybe from a database or from maybe a server API area, but we're going to be bringing in data into our HTML through an array object.

00:22 And so it's a little different from maybe real life, but the way we set it up will be the same whether it's coming from a database or another API or something like that.

00:35 We're just creating our own, um, object array. Instead. Alright, so let's go ahead and go back to our movies, HTML and CSS that we created previously, and we had put in two movies in article elements that were static.

00:54 Meaning they're not coming in dynamically, we just kind of hard-coded in each of those values, and if we had 10 movies, we would have had to hard-code 10 article elements in here, but we're going to make this really dynamic so that when anything changes, any of the movies or any movies get added or taken

01:12 out, it's going to automatically come in dynamically and update our page according to the updated information coming in. this will be a- a really good real life example for us to go through.

01:25 Alright, so first of all, we're going to take out the static information. And I'm going to go ahead and just delete one of them completely.

01:33 One of the movies that we had. And then I'm going to take this whole article and I'm just going to- head it with a Commander Control X so that we can create it as a template literal in our JavaScript file.

01:47 So I'll go ahead and create that JavaScript file and I'll give it the same name, movies.js. And I'll go ahead and paste that in.

01:56 Now that's definitely going to be in a template literal, so we want to go ahead and store that as such.

02:02 So I'm just going to call it HTML. And then put those back ticks, which remembers under the escape key. On either side of their end of that.

02:11 So now we've got kind of this big string, this template literal that's going to be stored in a variable called HTML.

02:19 Okay, so it's not coming into our HTML yet. But I'll go ahead and save that. And I also want to.

02:26 Hook up that new JavaScript file. I'm going to defer it. Right, so that. It can see that JavaScript. And so we can kind of test it out.

02:43 It won't be the final product, but we can kind of test it out by targeting this now that we have this empty section with no articles in it at all.

02:52 It has the ID of movie dash. We could set that up. And we could say the movie list is going to be, and then grab that from the Dom, the query selector.

03:07 And it was movie. An idea of movie dash list. And then we could come in here and just say, hey, grab this movie.

03:21 And in that movie, that's the empty section that we have. Umm, go ahead and put its inner HTML to be equal to this whole template literal.

03:35 Okay, and then if we look at that page, we can see that one movie coming in. Okay, it is coming in kind of dynamically, but it's still pretty static in JavaScript just because it's just the one title, the one, you know, image.

03:53 Edge. The one date. So it didn't do a whole lot, uhm, to change what we really had before. So now we're gonna go ahead and start using that template, array method, the dot for each array method to go through each.

04:08 And put it onto the screen. So if you look at your ponder here, we want it to come in and have these different, these four different movies coming in.

04:18 And they are stored in this, uhm, object array. So I'm gonna go ahead and cut. Copy that. We already have this, um, it is already here.

04:29 So I'm not gonna worry about that. We'd already had it from last week. But this is in the JavaScript area.

04:36 We're gonna go ahead and copy that. and I'll go ahead, and put it above this information. And we can see each of our movies has different information in each of these objects.

04:50 Okay, it does, it did have a little starter of a for each for each. And it wants to go ahead and put the code to create each movie in here.

04:58 So we can go ahead and just move this entire thing inside our object rate. So what's happening here is it's saying movie.

05:11 And if we look at our object array, we can see that it's called movies. So it's saying, hey, take this array, and for each item in the array, or each of these objects in the array, do this to it.

05:25 And instead of the word item, we're using movie. So just same, just with it on an S. And we are using an arrow function.

05:34 So a little bit different, uh, syntax here. But it's gonna go through and it's gonna do each movie. Now, to make this fit each movie, we need to, do what are called, um, expressions in our template literal.

05:46 Or expression interpolations is sometimes what they're called. So we're gonna go ahead and everywhere, it's gonna be different for each movie.

05:55 For example, this is the title. We can see that we have a m- movies dot title. So we're gonna say the name of the array dot the property.

06:05 And it's going to be placed in here, kind of like a variable. And remember, we need to use, uh, dollar sign and curly braces for each one of those.

06:14 They're coming in. As movie, that's why we use movie without an S, dot title. And so we can always look here and see exactly how that spelled.

06:24 And we'll go ahead and put title there. Again, we're gonna have to do the same thing for each image. And we can see that the, The image, um, SRC already has quote marks.

06:37 So we can go ahead and get rid of all this with the quote marks. And put dollar sign, movie. dot image SRC.

06:51 And I'm getting that from right here. Just exactly how it's spelled there. And the alternative text is the same way.

06:58 It's time. Each object comes in bringing the image out that goes with that. Same with the date. And again, just.

07:10 . Double check that that's the right spelling. And we also have an age. And we also have a genre. Turn around.

07:27 And then this one is interesting because we want to get the length of how many stars it has. So right here where we have the aria that the accessibility.

07:37 This five will be different for each one. And so we're actually going to do the Movie. Dot. Stars. Which is what it's called right here.

07:51 Stars. But we're going to get the length of that. Yeah, that's it. Okay, so it's going to see how many are in there.

08:02 And so it'll, it'll give us a four instead of just the movies. Or just the stars. Alright, and then right here the stars will be different for each one.

08:11 So we'll go ahead. And change this. And then the description will, of course, be different for each one. Okay, and then just make sure I spelled that right.

08:28 Description, yep. Okay. Alright, so now it should come through and show, Movies, and if we look at that, we get the last movie.

08:39 And that's happening because each time we're going through each movie, it's gonna go through Spider-Man, and then it's gonna go through the next one.

08:46 And and we're totally replacing each time that one. The HTML each time. So we would have to put something like a plus equal to saying add to what was already there instead of just replacing the whole section with a new article.

09:02 And then replacing each one each time. So then that will bring in all, all of them. Okay, so we're getting some dynamic data now, which is really cool.

09:12 So for each of these objects, it's going to go through and it's going to make sure it gets all the.

09:18 Information for each one of those and it will be different for each. Oh, I'm getting an undefined here though, which tells me I messed up on age.

09:29 And it's called ages with an S, not just age. I've just seen the dots there. Maybe you saw that. Okay.

09:37 That looks better. Alright. So, one thing that's maybe not as efficient as it could be here is because when we use a plus equal sign, it's .

09:49 . . We're having to update the DOM every time it loops through. So, we need to be a little more efficient so that the browser doesn't have to reparse and rebuild the entire container at every iteration.

10:01 So, this . . I mean, you could leave it this way. Um, it might . just take a little longer to load. But there is a little bit more efficient way I want to show you, So, what we're going to do with this next one is just change it a little bit here.

10:17 going to take out . Targeting this every single time. I don't think, whoops, I don't think we're going to put that outside of the lube.

10:26 And I'm going to go ahead and create this article. Instead of bringing it in as part of the template literal.

10:36 So I'm going to go ahead and just take out the opening and closing article tag. And this time I'm going to create it using JavaScript.

10:44 So I'm going to have a, uhm, constant called r- article. And I'm going to go ahead and create that article tag with a create element method.

10:59 And it needs to have a parameter of what am I creating? I'm creating an article. article. tag. And since it did have a class, if you noticed that, I'm going to go ahead and add that class back in, since that was part of the temporal literal before with the class equal.

11:14 And put it here. Did have a class of movie if you don't remember what the- that was. It was coming in and, you know, centering it and all those different stuff here.

11:26 So that's what we're doing here. Making sure it's got that class for CSS. Okay. And now, we're going to go ahead and- hand this HTML into this article.

11:38 So a little bit different down here. So now we're going to say, article inner HTML is going to be assigned.

11:48 The HTML take off the plus. And then we're going to append it onto the empty section with the idea of movie list.

12:02 And just so it's not putting a lot of that repar- parsing and rebuilding of the DOM, every loop, this is going to help it be a little more efficient.

12:12 And we'll see it should work the same way. and so instead of having that article inside of our tempo literal, we went ahead and created it, gave it a class, and then it- pended instead of doing that plus equal to it, which was a little less efficient.

12:26 But the same- same- same idea. Alright, so there we have some dynamic data coming in. Remember, in real life, this might be coming from a database or somebody- else's server with you using an API and fetching some data and converting that to a- a JavaScript, umm, object.

12:48 But this- it gives you the- I- you know, the whole kind of outline of- how that's going to work and using these expressions inside our template literal, these interpolations to go ahead and do that.

13:03 This is very similar to what you'll be doing with the book blog for your assignment this week.