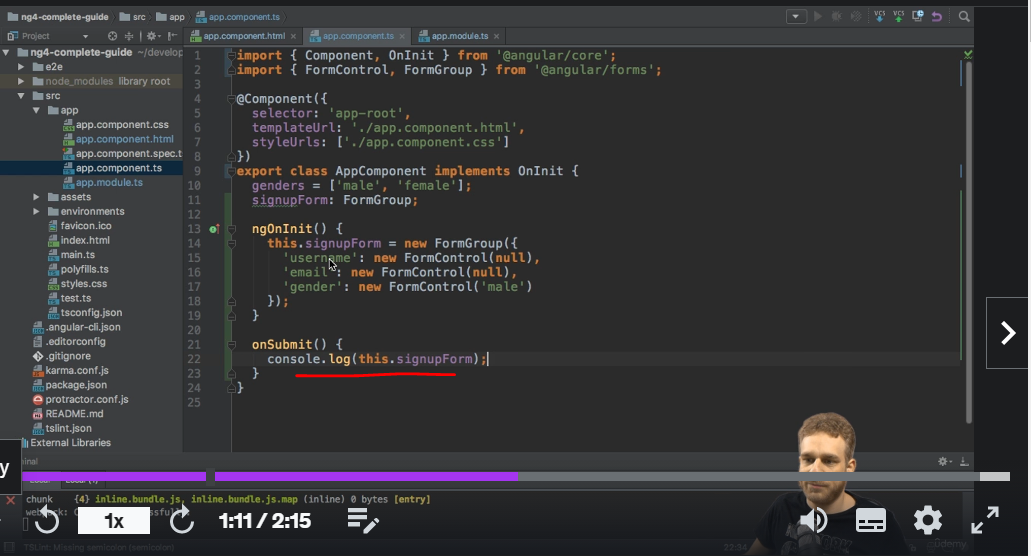
* -: In the last lecture we ***successfully synchronized our HTML form and our TypeScript form.***
* Let's now have a look at how we can **submit this form**.
* In the template driven approach, we used NG Submit, the NG submit directive here on this form element.
* Well, we still do the same here because we still want to react to this default submit event which is fired by html, by JavaScript.
* So we still add engine submit here, and we could execute our on submit method, any method you like.
* So let's add this method here now.
* **onSubmit.**
* The **difference** to the template driven approach now is that ***we don't need to get the form via this local reference.***
* That actually wouldn't work anymore because we're not using angulars auto creation mechanism.
* But we don't need to get this reference because, hey we created the form on our own.
* We already got access to it here in our types of code.
* Actually in every method in the TypeScript code.
* So we can simply console lock this sign up form just like this.

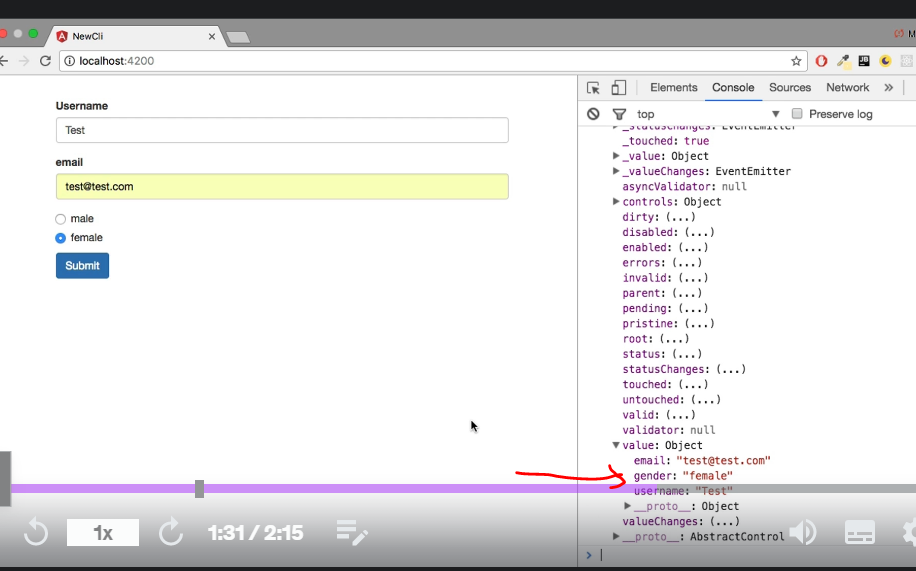


* And this is how we get access to the form.
* Now, if we go back to our running project and we simply enter something here, switch just to female to see if it is correctly updated and hit submit, we see our form group here.

Text

Description automatically generated

* And it has the properties we already know amongst them the value which correctly represents the object we passed here.



* Key value pairs with username, email, and gender and the appropriate values.
* This is the cool thing about the reactive approach.
* Whatever you set up here as an argument whatever you set up here as an object you pass to the form group, which makes up your form.
* That is what you get out as a value of the form.
* So you can bind it to your own model, the model of your application, and easily make sure that the form structure matches the structure of your model.
* So this is how you can submit the form how you can still access the value as you did before but now using your own form the form you created in Typescript.